

Convergence Analysis of Mixed Timescale Cross-Layer Stochastic Optimization

Junting Chen *Student Member, IEEE* and Vincent K. N. Lau, *Fellow, IEEE*

Dept. of Electronic and Computer Engineering

The Hong Kong University of Science and Technology

Clear Water Bay, Kowloon, Hong Kong

Email: {eejtchen, eeknlau}@ust.hk

Abstract

This paper considers a cross-layer optimization problem driven by multi-timescale stochastic exogenous processes in wireless communication networks. Due to the hierarchical information structure in a wireless network, a mixed timescale stochastic iterative algorithm is proposed to track the time-varying optimal solution of the cross-layer optimization problem, where the variables are partitioned into short-term controls updated in a faster timescale, and long-term controls updated in a slower timescale. We focus on establishing a convergence analysis framework for such multi-timescale algorithms, which is difficult due to the timescale separation of the algorithm and the time-varying nature of the exogenous processes. To cope with this challenge, we model the algorithm dynamics using stochastic differential equations (SDEs) and show that the study of the algorithm convergence is equivalent to the study of the stochastic stability of a virtual stochastic dynamic system (VSDES). Leveraging the techniques of Lyapunov stability, we derive a sufficient condition for the algorithm stability and a tracking error bound in terms of the parameters of the multi-timescale exogenous processes. Based on these results, an adaptive compensation algorithm is proposed to enhance the tracking performance. Finally, we illustrate the framework by an application example in wireless heterogeneous network.

Index Terms

Mixed timescale, Convergence analysis, Stochastic approximation, Cross-layer, Convex optimization

Convergence Analysis of Mixed Timescale Cross-Layer Stochastic Optimization

I. INTRODUCTION

Cross-layer resource optimization plays a critical role in the radio resource management of modern wireless systems. In existing literature, cross-layer optimization can be divided into two categories. When the system states are slowly varying, it is desirable to have dynamic controls adaptive to the instantaneous realizations of system state. For example, in [1], [2], the authors considered dynamic power control (adaptive to the instantaneous channel fading state) in wireless ad hoc networks. In [3], [4], adaptive joint beam-forming is considered to mitigate interference in a cellular network. In practice, it is quite difficult to obtain real-time observations of the global system states or real-time signaling message passing in a large scale network because of the signaling latency¹. As a result, it is desirable to adapt the control actions to the system state statistics instead of real-time realizations. For example, in [5] the author developed a limited feedback technique that utilizes the channel distribution information (CDI) for communication in multiuser MIMO beamforming networks. In [6], [7], the problem of robust transmit beamforming in multi-user communication system using the covariance-based channel information is considered.

In practice, system states usually evolve in mixed timescales in wireless networks. For example, in MIMO fading channels, the channel matrix changes in a short timescale (such as 10 ms) but the correlation and the path loss change in a longer timescale (such as minutes) [8], [9]. Another example of mixed timescale state evolution is between the queue length process (slower timescale) and the instantaneous link quality (faster timescale) [10], [11]. When the system has a multi-timescale state evolution, it is necessary to partition the controls in different timescales based on the *information structure* induced by the system topology. As an illustration, consider a wireless heterogeneous network with a macro base station (BS) and some relay BSs (RSs) as depicted in Fig. 1. The users transmit data flows to the macro BS with the assistance of the RSs. The control strategies depend on the channel state which evolves in mixed timescales. Suppose we want to adopt a cross-layer control to the flow data rate \mathbf{r} for each user and the transmission power \mathbf{p} on each link so as to maximize the average throughput in the example wireless network with time-varying channels. As the control policy may involve network-wise coordinations, one good strategy is to partition the control variables into local power control \mathbf{p} and global flow control \mathbf{r} , where the power control \mathbf{p}

¹For example, the X2 interface in e-Node B of LTE systems has latency of 10ms or more.

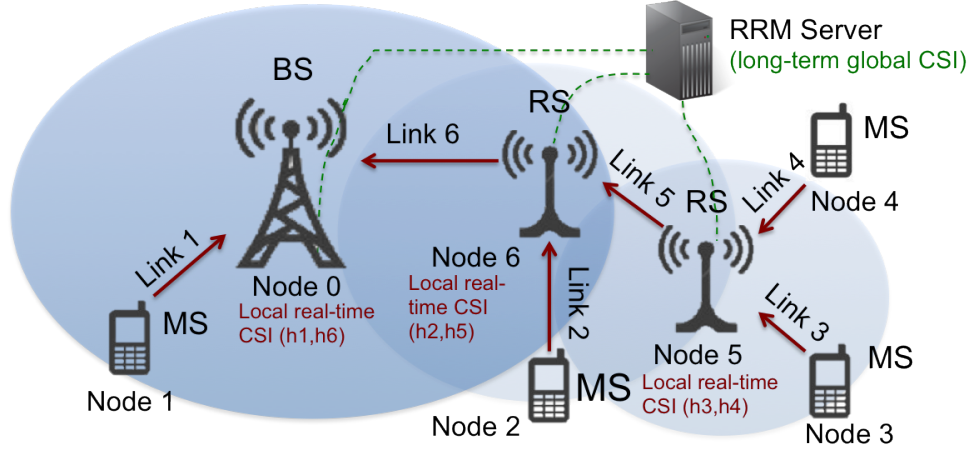


Figure 1. The topology of a wireless relay network with a radio resource management (RRM) server. The BSs and the RSs have local real-time CSI, while the RRM server has the long-term global CSI.

adapts to the instantaneous channel state information (CSI) locally and the flow rate r adapts to the CSI statistics with a global coordination. This is because, while it is realistic for each wireless node to acquire real-time local CSI, it is extremely difficult for the network controller to acquire real-time global CSI. If one considers pure fast timescale control for both the power and the flow data rate (such as in [2], [12]), the policy obtained will require real-time global CSI. This is difficult to achieve in practice and the system performance will be very sensitive to the signaling latency in the acquisition of global CSI. On the other hand, if one considers pure slow adaptation for both p and r (statistical adaptation), the resulting policy will fail to exploit the instantaneous transmission opportunity observed at each wireless node, and such an approach may be too conservative. Therefore, it is of great importance to have a complete cross-layer control framework with timescale separations that embraces the exogenous mixed-timescale state evolutions and exploits the information structure of the network topology.

There are quite a few works that studied controls with different timescale state evolutions [1], [13]–[17]. However, these works handled the different timescales separately in a heuristic manner². There are few works that considered a holistic cross layer optimization framework exploiting mixed timescale algorithms, not to mention the study of convergence properties of such mixed-timescale algorithms.

In this paper, we focus on the study of general cross-layer optimization for mixed timescales state processes. We first setup a stochastic optimization formulation to optimize an average network utility. The control variables are partitioned into a *short-term control* (adapts to fast timescale state processes)

²For example, the fast control and slow control are not optimizing the same optimization objective.

and a *long-term control* (adapts to slow timescale state processes) according to the information structure induced by specific network topology. These control variables are driven by mixed-timescale iterative algorithms to optimize the average network utility (objective function). An important question to the mixed timescale iterative algorithms is whether they will converge to the optimal solution. While the convergence of some iterative algorithms such as stochastic gradient [18], [19] are quite well-studied, these existing techniques considered one timescale iteration only and the convergence behavior of mixed timescale algorithms is highly non-trivial due to the mutual coupling between the short-term and long-term control variables. Specifically, there are several first order technical challenges that need to be addressed.

- **Coupling in the dynamics of the long-term and short-term iterations:** In most of the existing works involving multi-timescale control variables, the iterative algorithms driving different types of control variables are independent from each other. In other words, the intermediate iterates of the long-term variables in the outer-loop will not affect the convergence of the short-term variables in the inner loop. However, this decoupling is only justified when the short-term variables can converge to the optimal point arbitrarily fast or when we have closed-form solutions for the short-term variables. In general, we do not have closed-form solutions and each iteration in communication networks may also involve explicit signaling message passing. Hence, it may not be realistic to ignore the iteration dynamics in the short-term variables. Due to these couplings of the iterations between the long-term and short term variables, classical convergence proof in stochastic gradient [18], [19] or stochastic programming [20]–[22] cannot be applied.
- **Irreducible bias to the stochastic estimator:** In standard single timescale stochastic optimization with long-term control variables only, stochastic gradient [18], [19] is commonly used because the true gradient of the problem contains an expectation operator which does not have closed form expression in most cases. Using an *unbiased stochastic gradient estimator* [18], [19], we do not need to compute the true gradient in every iteration. If the original problem is strictly convex, the stochastic gradient algorithm will converge to the optimal solution [18], [19]. However, in the mixed timescale iterations, the convergence errors of the short-term control variables in the inner loop will induce an irreducible bias to the stochastic gradient estimator of the long-term control variables in the outer-loop. As such, standard convergence proof in stochastic gradient [18], [19] cannot be applied.
- **Exogenous Stochastic Variations of the State Processes:** In addition to the coupling in the mixed timescale iterations as well as the irreducible bias due to the short-term control iterations, the system states of the wireless system are also evolving stochastically. For example, the long-term state process (such as the channel path loss, the MIMO channel correlations or even the

network topology) may be time-varying due to the mobility of the users in the network or shadowing process. As such, these exogenous variations will also have a profound impact on the convergence behavior of the mixed timescale iterations.

A. Related words

In [23], [24], we have studied the convergence behavior of iterative algorithms in wireless systems under time-varying channel states. However, these works have focused on one-timescale iterations and the approach cannot be applied to address the above challenges in mixed-timescale iterations. There are also limited works on multi-timescale stochastic optimizations. In [20]–[22], multi-stage stochastic programming has been applied for logistic and financial planning problems. However, the problem considered has very a special form (linear program (LP)) and the solutions cannot be applied to our problems, which has a more general non-linear form. There are also some application examples [1], [13]–[17] that decompose the stochastic optimization problems into two timescale hierarchical solutions. However, all these works did not consider the tracking issues associated with exogenous variations of the problem parameters. In [25], [26], the authors studied the stochastic tracking algorithms in a regime-switching environment which is driven by a finite state two-timescale Markov chain. A dynamic step-size selection algorithm for the tracking in a regime-switching environment has been proposed in [27]. Nevertheless, a general understanding of the convergence behavior of mixed-timescale algorithms is still needed.

B. Our contribution

In this paper, we propose an analysis framework to study the convergence behavior of mixed timescale iterative algorithms of cross-layer stochastic optimization for wireless networks. Specifically, we first introduce the cross-layer stochastic optimization framework and the variable partitioning according to the information structure available. We then consider a mixed timescale iterative algorithm and study the dynamics and coupling using continuous time stochastic differential equation (SDE). We show that the study of the convergence behavior of the mixed timescale algorithm is equivalent to the study of *stochastic stability* of a virtual dynamic system specified by a system of SDE. Furthermore, the optimal solution of the stochastic optimization problem is equivalent to an *equilibrium point* of the virtual dynamic system. As a result, the convergence behavior of the algorithm is similar to the control problem of tracking a moving target [23], [24]. Based on this insight, we derive an upper bound on the tracking error of the mixed timescale algorithms under exogenous variations of the short-term and long-term state processes. Based on the insights of the impact of exogenous variations, we propose a low complexity compensation algorithm which can substantially enhance the tracking behavior of the mixed timescale algorithms. Finally, we apply this framework to an example topology in wireless

heterogeneous networks with relays. Numerical simulations verified the theoretical insights obtained and also demonstrated significant performance gain of the proposed compensation algorithms.

The paper is organized as follows. Section II illustrates the system model, where the cross-layer stochastic optimization framework and the mixed timescale stochastic approximation algorithm will be introduced. Section III studies the tracking behavior of the mixed timescale algorithm using the notion of virtual stochastic dynamic system (VSDS) and the techniques of Lyapunov stochastic stability. In Section V, a novel compensation algorithm is proposed to enhance the tracking performance. An application example on flow control and power allocation in wireless relay network is illustrated in Section VI. Section VII gives the numerical results, and we conclude the work in Section VIII.

Notations: For a scalar-valued function $F : \mathbb{R}^n \mapsto \mathbb{R}$, F_x denotes the vector of its partial derivatives with respect to vector $x = (x_1, \dots, x_n)$, i.e., the i -th component of F_x is $F_x^{(i)} = \frac{\partial F}{\partial x_i}$. For a vector-valued function $G : \mathbb{R}^n \mapsto \mathbb{R}^m$, G_x denotes the Jacobian matrix of $G = (G^{(1)}, \dots, G^{(m)})$, which is the partial derivatives of $G(\cdot)$ with respect to vector x , and whose (i, j) -th element is given by $\frac{\partial G^{(i)}}{\partial x_j}$. The notation $\lfloor x \rfloor$ denotes the largest integer that is no greater than x . For column vectors $x = [x_1 \dots x_n]^T$ and $y = [y_1 \dots y_m]^T$, $(x, y) = [x_1 \dots x_n y_1 \dots y_m]^T$ denotes a column vector that stacks the vectors x and y .

II. SYSTEM MODEL

In this section, we first introduce the cross-layer stochastic optimization framework with mixed timescale exogenous random processes. We then partition the control variables into short-term and long-term controls and describe the mixed timescale iterative algorithms to track the optimal solutions of the stochastic optimization problem. Based on that, we elaborate the convergence issues induced by the exogenous random processes and formally define the tracking error between the algorithm outputs and the optimal solution.

A. A Cross-Layer Stochastic Optimization Formulation

1) Network and Mobility Model: We first discuss the system model in a wireless communication network with node mobilities. Consider a wireless network with N_s static nodes and N_m mobile nodes. The location of the static nodes are fixed, and the mobile nodes are randomly distributed at time $t = 0$. For time $t > 0$, the mobile nodes change their locations according to a widely adopted *Levy walk mobility model* [28]–[30] described as follows.

Assumption 1 (Levy walk mobility model): Starting from time $t = 0_+$, each mobile node chooses a random destination in a restricted region centered at the initial location and moves at a constant speed in $(0, v_{\max}]$. Upon reaching the destination, the node pauses for some time and randomly chooses a

new destination and speed to go on. The travel distance and pause time at each step follow a truncated Levy distribution [29], [30]. ■

Nodes are inter-connected through wireless links, and since the node mobility is restricted, we assume that the network topology is fixed despite the mobility of the mobile nodes. The network topology can be represented by a directed graph $\mathcal{G} = (\mathcal{N}, \mathcal{L})$, where \mathcal{N} is the set of nodes and \mathcal{L} is the set of wireless links that connect the transmitters and the receivers. Fig. 1 illustrates an example wireless network, where \mathcal{N} is the set of BS, RSs and mobile users, and \mathcal{L} is the set of wireless transmission links between them.

Define the CSI for the j -th link as $h_j \in \mathcal{H}$. We adopt a fading model to the CSI h_j as $h_j = h_j^l h_j^s$, where $h_j^l = c_0 D_j^{-\iota} \in \mathcal{H}^l$ is the long-term CSI for the large-scale fading with c_0 being an antenna-gain-related constant, and $h_j^s \in \mathcal{H}^s$ is the short-term CSI for the small-scale fading [9], [31]. $D_j \geq D_{\min}$ is the distance between the j -th link, and ι is the path loss exponent.

2) *The CSI Dynamics:* We specify the dynamics of the CSI h_j^s and h_j^l by the exogenous random processes defined below.

Definition 1 (Exogenous stochastic processes): The short-term and long-term CSI processes $h_j^s(t)$ and $h_j^l(t)$ are driven by the following stochastic differential equations (SDE):

$$dh_j^s = -\frac{1}{2}a_H h_j^s dt + \sqrt{a_H} dW_t, \quad h_j^s(0) = h_0^s, \forall j = 1, \dots, N, \quad (1)$$

$$dh_j^l = -c_0 \iota D_j(t)^{-\iota-1} v_j(t) dt, \quad h_j^l(0) = h_0^l, \forall j = 1, \dots, N, \quad (2)$$

where $a_H > 0$, v_j is the relative speed of the j -th link, and W_t is a standard Brownian motion. ■

The positive parameter a_H specifies the time-correlation of the short-term exogenous processes $\{h_j^s(t)\}$, which are specified by the Ornstein-Uhlenbeck processes. $h_j^s(t)$ has a Gaussian stationary distribution and $|h_j^s|$ follows a Rayleigh distribution [8], [9]. On the other hand, defining $\epsilon(D_{\min}, v_{\max}) \triangleq 2c_0 \iota D_{\min}^{-\iota-1} v_{\max}$, we have $|dh_j^l| \leq \epsilon(D_{\min}, v_{\max}) dt$. The parameter ϵ , which is typically very small, represents the timescale of the long-term processes $\{h_j^l(t)\}$.

3) *Control Variables and the Problem Formulation:* We assume the following information structure for the wireless communication network.

Assumption 2 (Signaling information structure): Each node $k \in \mathcal{N}$ has knowledge of the *local* CSI (short-term and long-term) (h_j^s, h_j^l) for all the links $j \in \mathcal{L}$ that connect to node k . On the other hand, only the global long-term CSI $h^l = (h_1^l, \dots, h_{N_L}^l)$ is known at the central controller of the network. ■

For example, in Fig. 1, node 5 only has the local CSI knowledge of h_3 , h_4 and h_5 . Meanwhile, the RRM server has the long-term global CSI h_j^l for all the nodes j . Note that the above assumption is quite reasonable, because in practical wireless communication networks, it is relatively easy for each

node to acquire local real-time CSI but it would be difficult for the network controller to acquire the global real-time CSI.

According to the information structure assumption, the following defines the mixed timescale controls in the cross-layer optimization framework.

Definition 2 (Mixed Timescale Controls): The system has two sets of control variables, namely the *short-term control* and the *long-term control*. The short term control is denoted by a policy Ω^s which maps the realization of CSI vector $h = (h_1, \dots, h_N)$ to an action $\theta_x \in \mathbb{R}_+^{N_x}$. Similarly, the long-term control is denoted by a policy Ω^l which maps the realization of $h^l = (h_1^l, \dots, h_N^l)$ to an action $\theta_y \in \mathbb{R}_+^{N_y}$. ■

For example, the short-term control $\theta_x(t) = \Omega^s(h(t))$ and the long-term control $\theta_y(t) = \Omega^l(h^l(t))$ may correspond to the transmit power control on each wireless link and the flow control of a wireless network, respectively. The mixed timescale cross-layer stochastic optimization problem is given as follows.

Problem 1 (Mixed timescale cross-layer stochastic optimization problem):

$$\mathcal{P}_0(a_H, \epsilon) = \max_{\Omega^l, \Omega^s} \mathbb{E}[F(\theta_x, \theta_y; h(t))] \quad (3)$$

$$\text{subject to } w_i(\theta_x, \theta_y; h(t)) = 0, \quad i = 1, \dots, W_e \quad (4)$$

$$w_i(\theta_x, \theta_y; h(t)) \leq 0, \quad i = W_e + 1, \dots, W \quad (5)$$

$$q_j(\theta_y; h^l(t)) \leq 0, \quad j = 1, \dots, J, \forall h^l(t). \quad (6)$$

■

\mathcal{P}_0 can be decomposed into a family of *inner problems* and *outer problems*:

Problem 2 (The inner problem): For given h^s and h^l ,

$$\mathcal{P}_1(\theta_y, h^s, h^l) = \max_{\theta_x \geq 0} F(\theta_x, \theta_y; h^s, h^l) \quad (7)$$

$$\text{subject to } w_i(\theta_x, \theta_y; h(t)) = 0, \quad i = 1, \dots, W_e \quad (8)$$

$$w_i(\theta_x, \theta_y; h(t)) \leq 0, \quad i = W_e + 1, \dots, W \quad (9)$$

■

Problem 3 (The outer problem): For given h^l ,

$$\mathcal{P}_2(h^l) = \max_{\theta_y \geq 0} \mathbb{E}[\mathcal{P}_1(\theta_y, h^s, h^l) | h^l] \quad (10)$$

$$\text{subject to } q_j(\theta_y; h^l(t)) \leq 0, \quad j = 1, \dots, J. \quad (11)$$

■

The collection of outer-problem solution $\theta_y^*(h^l)$ for each h^l in (10)-(11) gives the optimal long-term policy Ω^{l*} . Similarly, the collection of the solution $\theta_x^*(h)$ for the inner problem $\mathcal{P}_1(\theta_y^*, h^l, h^s)$ for a given (h^s, h^l) gives the short-term optimal policy Ω^{s*} .

We have the following assumption on the optimization problem \mathcal{P}_0 .

Assumption 3 (Properties of \mathcal{P}_0): We assume the following for the problem \mathcal{P}_0 ,

- *Convex domain:* The constraint domain specified by (4)-(6) is convex.
- *Concave objective:* Given any CSI variable $h \in \mathcal{H}$, the objective function $F(\theta_x, \theta_y; h)$ is strictly concave over $(\theta_x, \theta_y) \in \mathbb{R}_+^{N_x} \times \mathbb{R}_+^{N_y}$.
- *Smoothness of $\theta_y^*(h^l)$:* Define an implicit mapping $\psi : h^l \mapsto \theta_y^*$ from the long-term CSI h^l to the optimal solution $\theta_y^*(h^l)$, i.e., $\psi(h^l)$ solves the outer problem $\mathcal{P}_2(h^l)$ in (10). There exists a constant $\varpi < \infty$, such that $\|\frac{\partial}{\partial h^l} \psi(\xi) - \frac{\partial}{\partial h^l} \psi(\xi')\| \leq \varpi \|\xi - \xi'\|$ for all $\xi, \xi' \in \mathcal{H}^l$.

■

Note that in the problem \mathcal{P}_0 , we may exclude equality constraints. However, one may always eliminate the equality constraints by either substituting them into the objective function or using the Lagrangian primal-dual method [32], [33]. Moreover, the last assumption ensures that there is no jump in the optimal solution $\theta_y^*(h^l)$ along the long-term CSI h^l .

Due to Assumption 3, there exists a unique optimal solution $(\Omega^{s*}, \Omega^{l*})$ for the problem \mathcal{P}_0 .

A strong motivation to such control variable partitioning is due to the information structure of local real-time and global real-time CSI observations in wireless networks. Due to the latency involved in global signaling of wireless networks, it is not scalable to adapt all the control variables to the fast varying short-term CSI h^s . On the other hand, adapting the control only to the slow varying long-term CSI h^l would be too conservative because it fails to exploit the real-time local CSI observations $h^s(t)$ for opportunistic diversity gain [31]. One favorable way to strike a balance between performance and scalability is to partition the control variables as long-term y and short-term control $\theta_y(t) = \Omega^s(h(t))$. Another motivation of the control timescale separation is the layered control architecture widely adopted in the wireless system [1], [34]. For example, the short-term control policy may correspond to physical layer control (e.g. power adaptation) and the long-term control may correspond to upper layer control (such as routing and admission control).

B. Example: Power and Flow Control in Wireless Relay Network

We illustrate the mixed timescale control with an example network with a macro BS and N_R RSs. A set of end users \mathcal{E} each transmits one data flow to the macro BS with the assistance of a collection of RSs \mathcal{R} . Fig. 1 illustrates a specific network topology with $N_R = 2$ RSs, $N_m = 4$ mobile users, $N = 6$ links, and $\mathcal{L} = \{1, \dots, 6\}$, $\mathcal{E} = \{1, 2, 3, 4\}$, $\mathcal{R} = \{5, 6\}$, $\mathcal{N} = \mathcal{E} \cup \mathcal{R} \cup \{0\}$, where node 0 denotes the macro BS. Wireless links towards a common receiving node share the same

time-frequency resource and multi-user detection (MUD) is used to handle cross-link interference. The maximum achievable transmission data rate at receiving node m is a set of rates c_k that satisfy the following constraints [31],

$$\sum_{k \in \mathcal{S}} c_k < \log \left(1 + \sum_{k \in \mathcal{S}} |h_k|^2 p_k \right), \quad \forall \mathcal{S} \subset \mathcal{L}^+(m) \quad (12)$$

where $\mathcal{L}^+(m)$ denotes the set of links that inject data flows to the receiving node m . For example, in Fig. 1, $\mathcal{L}^+(5) = \{3, 4\}$, $\mathcal{L}^+(6) = \{2, 5\}$ and $\mathcal{L}^+(0) = \{1, 6\}$.

Denote r_j as the flow rate from node $j \in \mathcal{E}$ and $\Omega^r(\mathbf{h}^l)$ as the flow control policy that maps the large-scale fading variable $\mathbf{h}^l = (h_1^l, \dots, h_N^l)$ to the flow control $\mathbf{r} = (r_1, \dots, r_{|\mathcal{E}|})$. Denote p_k as the power allocation on link $k \in \mathcal{L}$ and $\Omega^p(\mathbf{h})$ as the power allocation policy that maps the CSI vector $\mathbf{h} = (h_1, \dots, h_N)$ to the transmission power $\mathbf{p} = (p_1, \dots, p_N)$. Corresponding to (3), a two-timescale stochastic maximization can be formulated as follows,

Problem 4 (Power and flow control in wireless relay network):

$$\max_{\Omega^r, \Omega^p} \quad \mathbb{E} \left[\sum_{j \in \mathcal{E}} \log(r_j) - V \sum_{k \in \mathcal{L}} p_k \right] \quad (13)$$

$$\text{subject to} \quad \sum_{k \in \mathcal{S}} c_k < \log \left(1 + \sum_{k \in \mathcal{S}} |h_k|^2 p_k \right) \quad \forall \mathcal{S} \subset \mathcal{L}^+(m), \forall m \in \mathcal{R} \cup \{0\} \quad (14)$$

$$c_k = \sum_{j \in \mathcal{C}(k)} r_j \quad \forall k \in \mathcal{L} \quad (15)$$

$$\sum_{k \in \mathcal{L}^+(m)} c_k - \sum_{k \in \mathcal{L}^-(m)} c_k = 0 \quad \forall m \in \mathcal{R} \quad (16)$$

where $\mathcal{C}(k)$ denotes the collection of data flows r_j that routes through link k , and $\mathcal{L}^-(m)$ denotes the set of links that carry data flows from the transmitting node m . ■

For example, in Fig. 1, $\mathcal{C}(4) = \{r_4\}$, $\mathcal{C}(5) = \{r_3, r_4\}$, $\mathcal{C}(6) = \{r_2, r_3, r_4\}$, and $\mathcal{L}^-(k) = k$, for $k = 1, \dots, 6$. We consider proportional fair utility [35] in the objective function of (13). Constraints (14)-(15) are the MUD capacity constraints, and constraints in (16) are the flow balance constraints, where the incoming data flow should be balanced with the outgoing data flow at each RS.

With timescale separation of the control Ω^p and Ω^r , the above stochastic maximization can be decomposed into families of inner problems and outer problems.

Problem 5 (Inner power control): For given flow data rate \mathbf{r} and CSI \mathbf{h} ,

$$Q(\mathbf{r}; \mathbf{h}) = \max_{\mathbf{p} \geq 0} \quad -V \sum_{k \in \mathcal{L}} p_k \quad (17)$$

$$\text{subject to} \quad \sum_{k \in \mathcal{S}} c_k(\mathbf{r}) < \log \left(1 + \sum_{k \in \mathcal{S}} |h_k|^2 p_k \right), \quad \forall \mathcal{S} \subset \mathcal{L}^+(m), \forall m \in \mathcal{R} \cup \{0\} \quad (18)$$

where $c_k(\mathbf{r}) = \sum_{j \in \mathcal{C}(k)} r_j$, $\forall k \in \mathcal{L}$. ■

Problem 6 (Outer flow control): For given the large-scale fading variable \mathbf{h}^l ,

$$\max_{\mathbf{r} \geq 0} \quad \mathbb{E} \left[\sum_{j \in \mathcal{E}} \log(r_j) + Q(\mathbf{r}; \mathbf{h}) \right] \quad (19)$$

$$\text{subject to} \quad \sum_{k \in \mathcal{L}^+(m)} c_k(\mathbf{r}) - \sum_{k \in \mathcal{L}^-(m)} c_k(\mathbf{r}) = 0 \quad \forall m \in \mathcal{R}. \quad (20)$$

Components in the example		Corresponding components in the model	
\mathbf{h}^s	Vector of small-scaling fading of the wireless links	h^s	Short-term CSI variable
\mathbf{h}^l	Vector of path loss variables of the wireless links	h^l	Long-term CSI variable
\mathbf{h}	The aggregated CSI	h	The aggregated CSI
Ω^p	Power control policy	Ω^s	Short-term control policy
Ω^r	Flow control policy	Ω^l	Long-term control policy
\mathbf{p}	Power allocation	θ_x	Short-term control variable
\mathbf{r}	Flow rate control	θ_y	Long-term control variable
(13)	Objective function	$F(\cdot)$	Objective function
(18)	Constraints for the inner problem	(6)-(4)	Constraints for the inner problem (7)
(20)	Constraints for the outer problem	(11)	Constraints for the outer problem (10)

Table I

PROBLEM ASSOCIATIONS BETWEEN THE EXAMPLE AND THE MIXED TIMESCALE SYSTEM MODEL.

■

In this example, the power allocation \mathbf{p} in (13) corresponds to the short-term control variable θ_x in (3), and the flow data rate variable \mathbf{r} in (13) corresponds to the long-term control variable θ_y . The objective function in (3) is specified by

$$F(\mathbf{p}, \mathbf{r}; \mathbf{h}^s, \mathbf{h}^l) = \sum_{j \in \mathcal{E}} \log(r_j) - V \sum_{k \in \mathcal{L}} p_k.$$

Meanwhile, the inner problem constraints (8)-(9) and the outer problem constraints (11) are specified by (18) and (20), respectively. Moreover, the short-term control policy Ω^s in Definition 2 is specified by the power control $\mathbf{p}(t) = \Omega^p(\mathbf{h}(t))$ in this example. Similarly, the long-term control policy Ω^l is specified by $\mathbf{r}(t) = \Omega^r(\mathbf{h}^l)$ here. One can check that the objective function (13) is concave in (\mathbf{p}, \mathbf{r}) and the optimization domain specified by (14)-(16) is convex. In addition, as the Lagrangian function [32], [33] of the constrained problem (13)-(16) is continuous in \mathbf{h} , the smoothness condition in Assumption 3 is satisfied as well. Table I summarizes the associations between the example and the mixed timescale system model in Section II-A.

The control variable partitioning is motivated by the information structure of the wireless relay network. On one hand, the CSI h_k is available at some corresponding receiving node $m = \{k : k \in \mathcal{L}^+(m)\}$. Hence the inner problem is solved locally at each receiving node $m \in \mathcal{R} \cup \{0\}$ based on real-time CSI $\{\mathbf{h}_k : \forall k \in \mathcal{L}^+(m)\}$. On the other hand, solving the outer problem with the flow balance constraints (16) (or (20)) requires a global coordination. Updating the variable \mathbf{r} needs to have the global knowledge of $Q(\mathbf{r}; \mathbf{h})$ and the statistics of the long-term CSI \mathbf{h}^l . It also needs to handle the global coupling induced by the flow balance constraints (16) (or (20)). As a result, explicit message passing is involved and the update can only be done in a longer timescale.

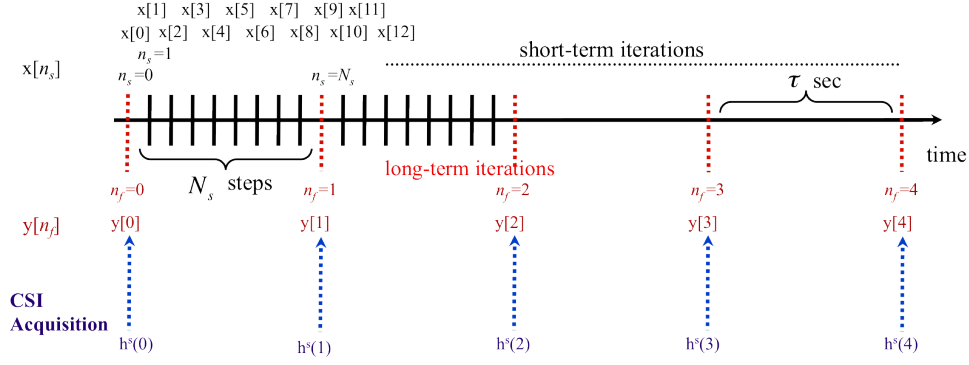


Figure 2. An illustration of the two-timescale algorithm.

C. Iterative Solution of the Mixed-Timescale Stochastic Optimization Problem

In this section, we discuss a stochastic gradient based two-timescale algorithm for solving $\mathcal{P}_0(a_H, \epsilon)$, which consists of an inner iteration and an outer iteration. We are interested in the case where the inner and outer problems \mathcal{P}_1 and \mathcal{P}_2 do not have closed form solutions and iterations are needed to find the optimal solution.

Let $x = (\theta_x, \lambda_x)$ be the variable for the inner iteration, where θ_x is the short-term control variable in (7) and λ_x is an algorithm specific auxiliary variable³. Similarly, let $y = (\theta_y, \lambda_y)$ be the variable for the outer iteration, where θ_y is the long-term control variable in (10) and λ_y is the auxiliary variable. The time is partitioned into *frames* with duration τ and slots as illustrated in Fig. 2. One frame consists of N_s slots and local real-time CSI is acquired at each node at the beginning of each frame. During the n_s -th slot and the n_f -th frame ($n_f = \lfloor n_s/N_s \rfloor$), the short-term variable x_{n_s} and long-term variable y_{n_f} are updated according to the following mixed-timescale iterations,

$$x_{n_s} = \mathcal{P}_{\mathcal{X}(y)} \left[x_{n_s-1} + \gamma_{n_s} G(x_{n_s-1}, y_{n_f}; h^s(n_f\tau), h^l(n_f\tau)) \right] \quad (21)$$

$$y_{n_f} = \mathcal{P}_{\mathcal{Y}} \left[y_{n_f-1} + \mu_{n_f} K(x_{n_s}, y_{n_f-1}; h^s(n_f\tau), h^l(n_f\tau)) \right] \quad (22)$$

where γ_{n_s} and μ_{n_f} are the step size sequences, $\mathcal{P}_{\mathcal{D}}[\cdot]$ is an Euclidian projection onto the domain \mathcal{D} , and $\mathcal{X}(y)$ and \mathcal{Y} are the domains related to the problem constraints (8)-(9) and (11). Fig. 2 illustrates the timescales of the iterations in (21)-(22). Each node acquires local CSI at each frame boundary and updates the short-term control variables x_{n_s} once at each slot according to (21). The centralized controller updates the long-term variable y_{n_f} once every frame according to (22).

In addition, we consider the following assumptions on the iterations (21)-(22).

³In Lagrange primal-dual methods, λ_x is the Lagrange multiplier.

Assumption 4 (Properties of the iterations): Denote $\mathcal{M}(\cdot) \triangleq (G(\cdot), K(\cdot))$ as the mapping of the joint iteration vector (x_{n_s}, y_{n_f}) . We assume the following properties:

- *Definiteness of the iteration mappings:* The Jacobian matrices of the iteration mappings $G_x(\cdot) \triangleq \nabla_x G(\cdot)$, $K_y(\cdot) \triangleq \nabla_y K(\cdot)$, and $\nabla \mathcal{M}$ have the following properties: There exists $\alpha_x, \alpha_y, \alpha > 0$, such that $x^T G_x x \leq -\alpha_x \|x\|^2$, $y^T K_y y \leq -\alpha_y \|y\|^2$, and $(x, y)^T \nabla \mathcal{M}(x, y) \leq -\alpha (\|x\|^2 + \|y\|^2)$ for all $(x, y) \in (\cup_{y \in \mathcal{Y}} \mathcal{X}(y)) \times \mathcal{Y}$, given any $(h^s, h^l) \in \mathcal{H}^s \times \mathcal{H}^l$.
- *Lipschitz continuous and bounded growth:* There exist positive constants l_x and l_y , such that $\|K(x, y_1; \cdot) - K(x, y_2; \cdot)\| \leq l_y \|y_1 - y_2\|$, for all y_1 and y_2 , and $\|K_x(x, y; \cdot)x\| \leq l_x \|x\|$, for all x .

■

Note that the first assumption is easily satisfied in a compact domain. It is needed to guarantee the iterations (21)-(22) to be stable under exogenous variation of $h(t)$, where the parameters α_x and α_y indicate the convergence speed of the inner iteration and the outer iteration, respectively, and α indicates the convergence speed of the whole algorithm (under one-timescale). The second assumption is a standard assumption for studying the convergence. In this paper, we are interested in the convergence of (21)-(22) to the stationary points defined as follows.

Definition 3 (Stationary points of (21)-(22)): Given y and (h^s, h^l) , the partial stationary point $\hat{x}(y; h^s, h^l)$ of (21) is given by the solution of

$$x - \mathcal{P}_{\mathcal{X}(y)} \left[x + G(x, y; h^s, h^l) \right] = 0.$$

Similarly, given h^s and h^l , the stationary point $(x^*(h), y^*(h^l))$ of (21)-(22) is given by the solution of

$$\begin{aligned} x - \mathcal{P}_{\mathcal{X}(y)} \left[x + G(x, y; h^s, h^l) \right] &= 0 \\ y - \mathcal{P}_{\mathcal{Y}} \left[y + \mathbb{E} K(x, y; h^s, h^l) \right] &= 0 \end{aligned}$$

■

Under Assumption 4, the partial stationary point and the stationary point defined above is unique.

Remark 1 (Interpretation of the projection): The projection $\mathcal{P}_{\mathcal{D}}(\theta)$ is to find the nearest point $x \in \mathcal{D}$ from θ , i.e., $x = \mathcal{P}_{\mathcal{D}}(\theta)$ is the solution to the minimization problem $\min_x \|x - \theta\|_2^2$, subject to $x \in \mathcal{D}$. Note that if the constraint set is a hyper-rectangle, i.e., $\mathcal{D} = \Pi_{i=1}^N [a_i, b_i]$, the projection can be computed by restricting each component as $a_i \leq x^{(i)} \leq b_i$. When a general constraint set \mathcal{D} is considered, the projection can be computed by Lagrange multipliers [32], [33].

■

Remark 2 (Examples of iterative algorithms): Different choices of the iteration mappings $G(\cdot)$ and $K(\cdot)$ yield different variants of the stochastic algorithm. We review a few commonly used algorithms in the following.

- *Stochastic projected gradient [18], [19]*: The mappings $G(\cdot)$ and $K(\cdot)$ are the gradients of the objective function $F(\cdot)$ in (3). Specifically,

$$G(x_{n_s-1}, y_{n_f}; h^s(t_{n_f}), h^l(t_{n_f})) = \Gamma_x \nabla_x F(x_{n_s-1}, y_{n_f}, h^s(n_f\tau), h^l(n_f\tau)) \quad (23)$$

$$K(x_{n_s}, y_{n_f-1}; h^s(t_{n_f}), h^l(t_{n_f})) = \Gamma_y \nabla_y F(x_{n_s}, y_{n_f-1}, h^s(n_f\tau), h^l(n_f\tau)) \quad (24)$$

where Γ_x and Γ_y are positive definite scaling matrices to accelerate the convergence. The iteration variables x_{n_s} and y_{n_f} in (21)-(22) correspond to θ_x and θ_y in (3), i.e., $\theta_x(n_s) = x_{n_s}$ and $\theta_y(n_f) = y_{n_f}$. In addition, the projection domains in (21)-(22) are specified by

$$\mathcal{X}(y) = \{\theta_x \in \mathbb{R}_+^{N_x} : (8) - (9) \text{ are satisfied}\}, \quad \mathcal{Y} = \{\theta_y \in \mathbb{R}_+^{N_y} : (11) \text{ is satisfied}\}. \quad (25)$$

Note that $K(\cdot)$ is a stochastic estimator of the desired gradient descent direction

$$\nabla_y \mathbb{E}[\max_x F(x, y_{n_f-1}; h(n_f\tau)) | h^l(n_f\tau)] \text{ (c.f. [19], [36]).}$$

- *Stochastic primal-dual algorithm [33], [37]*: We first form a Lagrangian function of the problem in (3),

$$L(\theta_x, \theta_y, \lambda_x, \lambda_y; h) = F(\theta_x, \theta_y; \cdot) - \sum_i \lambda_{x,i} w_i(\theta_x, \theta_y; \cdot) - \sum_j \lambda_{y,j} q_j(\theta_y; \cdot)$$

where $\lambda_x \geq 0$ and $\lambda_y \geq 0$ are the Lagrange multipliers for the primal variables θ_x and θ_y , respectively. Let $x = (\theta_x, \lambda_x)$ and $y = (\theta_y, \lambda_y)$. The mappings $G(\cdot)$ and $K(\cdot)$ for stochastic primal-dual algorithm are given by

$$G(\cdot) = \begin{pmatrix} \nabla_{\theta_x} L(\theta_x, \theta_y, \lambda_x, \lambda_y; h) \\ -\nabla_{\lambda_x} L(\theta_x, \theta_y, \lambda_x, \lambda_y; h) \end{pmatrix}, \quad \text{and} \quad K(\cdot) = \begin{pmatrix} \nabla_{\theta_y} L(\theta_x, \theta_y, \lambda_x, \lambda_y; h) \\ -\nabla_{\lambda_y} L(\theta_x, \theta_y, \lambda_x, \lambda_y; h) \end{pmatrix}. \quad (26)$$

In addition, the projection domains are given by

$$\mathcal{X}(y) = \mathbb{R}_+^{N_x} \times \mathbb{R}_+^W, \quad \text{and} \quad \mathcal{Y} = \mathbb{R}^{N_y} \times \mathbb{R}_+^J. \quad (27)$$

■

Under Assumption 4, the convergence of (21)-(22) can be established from standard techniques [19], [36] for static CSI h^s and h^l , as summarized below.

Theorem 1 (Convergence under static CSI h^s and h^l): Consider $G(\cdot)$ and $K(\cdot)$ are given by (23)-(24) (or (26)), with the projection domains \mathcal{X} and \mathcal{Y} given by (25) (or (27)). If the step size sequences γ_{n_s} and μ_{n_f} satisfy $\sum \gamma_{n_s} = \infty, \sum \mu_{n_f} = \infty$ and $\sum \gamma_{n_s}^2 < \infty, \sum \mu_{n_f}^2 < \infty$, then the iteration (x_{n_s}, y_{n_f}) in (21)-(22) converges to the stationary point $(x^*(h^s, h^l), y^*(h^l))$. Furthermore, $(\theta_x^*(h^s, h^l), \theta_y^*(h^l))$ solves the problem in (3). ■

However, when the CSI $h^s(t)$ and $h^l(t)$ are time-varying, the above convergence is not guaranteed. This is because, on one hand, the inner iteration x_{n_s} may not converge and hence induce bias to the estimator $K(\cdot)$ for updating y_{n_f} . On the other hand, the optimal target $y^*(h^l(t))$ is time-varying

as well, and existing convergence results [18], [19] fail to apply. In this paper, we shall focus on investigating the convergence behavior of the mixed-timescale iterations (21)-(22) when the CSI $h^l(t)$ and $h^s(t)$ have mixed-timescale stochastic time-variations.

D. Impact of Exogenous Variations and Tracking Errors

With the presence of the exogenous variations of $h(t)$, the mixed timescale iterations in (21)-(22) are continuously perturbed and the convergence to the stationary point target $(x^*(h(t)), y^*(h^l(t)))$ is not guaranteed. The impact of the exogenous variations can be summarized as follows,

- *Impact on the inner iteration x_{n_s}* : Since $h(t)$ is time-varying, x_{n_s} needs to track the time-varying optimal point $x^*(h(t))$. The convergence error may depend on the relative variation speed between the iteration dynamics (21) and the exogenous variations of $h(t)$.
- *Impact on the outer iteration y_{n_f}* : Likewise, the optimal point $y^*(h^l(t))$ is time-varying, and y_{n_f} may hardly reach $y^*(h^l(t))$. Moreover, the convergence error of the inner problem yields $\hat{\mathcal{P}}_1 - \mathcal{P}_1 \neq 0$, which induces a bias to the estimator $K(\cdot)$ in (22).

We formally define the tracking error for the iterations (21)-(22).

Definition 4 (Tracking error): The mean square tracking error for the short-term control variable x is defined as

$$e_x = \lim_{n_f \rightarrow \infty} \sup_{n_f} \frac{1}{n_f} \sum_{i=1}^{n_f} \mathbb{E} \left[\|x_{iN_s} - \hat{x}(y_i, h^s(i\tau), h^l(i\tau))\|^2 \right] \quad (28)$$

whereas, the tracking error for the long-term control variable y is defined as

$$e_y = \lim_{n_f \rightarrow \infty} \sup_{n_f} \frac{1}{n_f} \sum_{i=1}^{n_f} \mathbb{E} \left[\|y_i - y^*(h^l(i\tau))\|^2 \right]. \quad (29)$$

■

In the rest of the paper, we shall study the tracking errors e_x and e_y under the mixed-timescale CSI $h^s(t)$ and $h^l(t)$.

III. VIRTUAL DYNAMIC SYSTEMS FOR CONVERGENCE ANALYSIS

In this section, we derive the *virtual dynamic systems* for studying the tracking error of the iterations (21)-(22) under time-varying CSI. We first consider a mean continuous time dynamic system (MCTS) which captures the mean behavior of the mixed-timescale iterations in (21)-(22) under static h^l . Using SDE approximations, we then extend the results to consider the impact of time-varying $h^l(t)$ on the overall tracking errors and derive the VSDS.

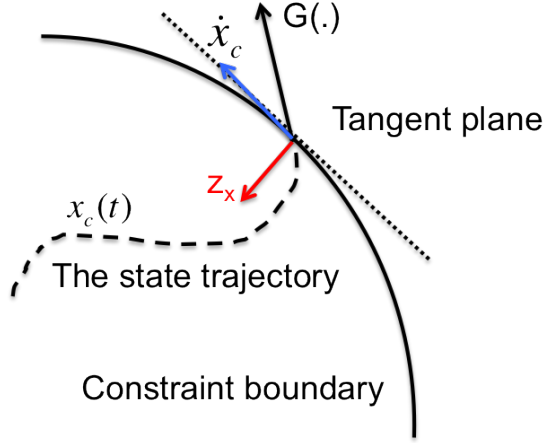


Figure 3. Illustration of the reflection term $z_x(t)$ when the virtual state trajectory $x_c(t)$ reaches the boundary of the constrain domain.

A. *Case 1: The Mean Continuous-Time Dynamic System (MCTS) for Static h^l and Time-varying $h^s(t)$*

In this subsection, we consider the case for static h^l and time-varying $h^s(t)$. Under time-varying $h^s(t)$, constant step size γ should be used in the inner iteration (21) to track the time-varying partial stationary point $\hat{x}(t)$. On the other hand, since the stationary point $y^*(h^l)$ is static, a diminishing step size μ_{n_f} can be used in the outer iteration (22) to assist the convergence. We derive a mean continuous-time dynamic system (MCTS) to characterize the “mean” behavior of the algorithm trajectories for (21)-(22). The MCTS is defined as follows.

Definition 5 (Mean continuous-time dynamic system (MCTS)): The state trajectory of mean continuous-time dynamic systems (MCTS) $x_c(t)$ and $y_c(t)$ are defined as the solutions to the following Skorohod reflective ordinary differential equations (ODEs) [38],

$$\dot{x}_c = G(x_c, y_c, h^s, h^l) + z_x \quad (30)$$

$$\dot{y}_c = k(y_c, h^l) + z_y. \quad (31)$$

where $\dot{x}_c \triangleq \frac{d}{dt}x_c(t)$, $\dot{y}_c \triangleq \frac{d}{dt}y_c(t)$. The terms $z_x(t)$ and $z_y(t)$ are the reflection terms to keep the trajectories x and y inside their domains $\mathcal{X}(y)$ and \mathcal{Y} , respectively. The function $k(\cdot)$ is defined as

$$k(y, h^l) \triangleq \lim_{n_f \rightarrow \infty} \mathbb{E}[K(\hat{x}(y, h(n_f\tau)), y, h(n_f\tau))] \quad (32)$$

where $K(\cdot)$ is the iteration mapping specified in (22). ■

Note that since the short-term CSI process $h^s(t)$ in (1) is ergodic and stationary, the limit in (32) always exists.

Remark 3 (Interpretation of the reflection): The reflection term is the minimum force to restrict the trajectory inside the constraint domain. Taking z_x for example, for $x_c \in \mathcal{X}$, in the interior of \mathcal{X} ,

$z_x = 0$. For $x \in \partial\mathcal{X}$, the boundary of \mathcal{X} , z_x lies in the convex cone generated by the inward normals on the surface of the boundary as illustrated in Fig. 3. The magnitude of z_x is such that $G(\cdot) + z_x$ lies in the tangent of the surface. Therefore, when the trajectory reaches the boundary, it can only go along on the boundary. Note that $z_x(t)$ and $z_y(t)$ can be computed using Lagrange multipliers when the constraint domain are not hyper-rectangles. Please refer to Appendix A for a derivation of the reflection terms. ■

In the following, we illustrate the connection between the algorithm trajectory (21)-(22) and the MCTS (30)-(31). We first derive the property of the *equilibrium* of MCTS.

Definition 6 (Equilibrium and partial equilibrium): The point (x_c^*, y_c^*) is an equilibrium of the MCTS (30)-(31) under $(h^s, h^l) \in \mathcal{H}^s \times \mathcal{H}^l$, if $\dot{x}_c = \dot{y}_c = 0$, i.e., $G(x_c^*, y_c^*, h^s, h^l) + z_x = 0$ and $k(y_c^*, h^l) + z_y = 0$. In addition, for any $y_c \in \mathcal{Y}$, $\hat{x}_c(y_c, h)$ is a partial equilibrium of the MCTS in (30) if $\dot{\hat{x}}_c = 0$, i.e., $G(\hat{x}_c, y_c, h^s, h^l) + z_x = 0$. ■

There is a strong connection between the iteration trajectory and the MCTS as summarized in the following theorem.

Theorem 2 (Connection between the algorithm trajectory and the MCTS): Assume $\epsilon = 0$ (static long-term CSI h^l). Consider $G(\cdot)$ and $K(\cdot)$ are given by (23)-(24) (or (26)), with the projection domains \mathcal{X} and \mathcal{Y} given by (25) (or (27)), and the step size sequences satisfy (i) $\gamma_{n_f} = \gamma$ for some small enough $\gamma > 0$, and (ii) $\sum \mu_{n_f} = \infty$ and $\sum \mu_{n_f}^2 < \infty$. In addition, the short-term CSI timescale is much slower than the algorithm timescale, i.e., $a_{HT} \ll \gamma$. Then the algorithm iteration trajectory $(x_{n_s(t)}, y_{n_f(t)})$ in (21)-(22) converges to the virtual state trajectory of the MCTS $(x_c(t), y_c(t))$ in (30)-(31) in probability, i.e., for any $\eta > 0$,

$$\limsup_{t \rightarrow \infty} \Pr \{ \|x_{n_s(t)} - x_c(t)\| > \eta \} = 0, \text{ and } \limsup_{t \rightarrow \infty} \Pr \{ \|y_{n_f(t)} - y_c(t)\| > \eta \} = 0$$

where $n_s(t) = \lfloor tN_s/\tau \rfloor$ and $n_f(t) = \lfloor t/\tau \rfloor$. ■

The theorem is established using stochastic approximation [18], [19]. We sketch the proof in Appendix B.

From Theorem 2, the study of the algorithm convergence is equivalent to the study of the *stability*⁴ of the MCTS. The MCTS in (30)-(31) can be viewed as the desired “mean” trajectory in the continuous-time counter-part of the discrete-time algorithm iterations (21)-(22), which has filtered the noisy perturbation induced by the exogenous variation of $h^s(t)$ in the estimator $K(\cdot)$. Hence, the stability of the MCTS provides a necessary and sufficient condition for the convergence of the original iterations (21)-(22) under static h^l .

⁴A deterministic dynamic system $\dot{x} = f(t, x)$ is asymptotically stable at the equilibrium x^* , if there is a $\delta > 0$, such that for any $\|x(0) - x^*\| \leq \delta$, $x(t) \rightarrow x^*$, as $t \rightarrow \infty$ [39].

Note that, under Assumption 4, the MCTS (22) is asymptotically stable [39], i.e., $y_c(t) \rightarrow y^*$ as $t \rightarrow \infty$. Therefore, from Theorem 2, we have the following result on the convergence under static h^l and time-varying $h^s(t)$ when the CSI timescale is sufficiently slower than the algorithm timescale.

Corollary 1 (Convergence under static h^l and slow time-varying $h^s(t)$): For sufficiently fast algorithm iteration, i.e., $a_H \tau \ll \gamma$, the iteration $(x_{n_s(t)}, y_{n_f(t)})$ in (21)-(22) converges to $(x^*(t), y^*)$ almost surely, i.e., $\|x_{n_s(t)} - x^*(t)\| \rightarrow 0$ and $\|y_{n_f(t)} - y^*\| \rightarrow 0$ almost surely, as $t \rightarrow \infty$. ■

Fig. 4.a) illustrates the relationships between the algorithm iterations, the virtual dynamic system MCTS and the moving equilibrium of the MCTS for the case when both h^s and h^l are static. The virtual system MCTS as well as the algorithm iterations x_{n_s} and y_{n_f} would eventually converge to the static equilibrium $(x^*(h), y^*(h^l))$. Fig. 4.b) illustrates the case for static h^l and slowly time-varying $h^s(t)$. For variable y , there is a static equilibrium $y^*(h^l)$ of the virtual dynamic system MCTS, and the virtual system state $y_c(t)$ converges to the static target $y^*(h^l)$. For variable x , the trajectory of the partial equilibrium $\hat{x}(y_c(t), h(t))$ is driven by the time-varying $y_c(t)$ and $h(t)$, and it eventually converges to the trajectory of the moving equilibrium $x^*(h(t))$, as $y_c(t)$ converges to $y^*(h^l)$. Meanwhile, the dynamics of the MCTS tracks the moving partial equilibrium $\hat{x}(y_c(t), h(t))$. For both x and y , the algorithm iterations x_{n_s} and y_{n_f} in (21)-(22) roughly follow the dynamics of the virtual system MCTS. However, the case is different under time-varying $h^l(t)$ and $h^s(t)$, as illustrated in Fig. 4.c). The equilibrium $y^*(h^l(t))$ of the virtual dynamic system MCTS moves around and $y_c(t)$ never converges. Affected by the time-varying $y^*(t)$ and the error gap $y_c(t) - y^*(t)$, the dynamics of the partial equilibrium $\hat{x}(y_c(t), h(t))$ cannot converge to the optimal trajectory $x^*(h(t))$. Nevertheless, the algorithm iterations x_{n_s} and y_{n_f} in (21)-(22) still follow the behavior of the virtual dynamic system MCTS.

In the next section, we extend the MCTS equivalence framework to the case with time-varying $h^l(t)$ and $a_H \tau \approx \gamma$.

B. Case 2: Virtual Stochastic Dynamic System for Time-Varying $h^s(t)$ and $h^l(t)$

When $h^l(t)$ is time-varying, the algorithm iterations (x_{n_s}, y_{n_f}) in (21)-(22) should continuously track the time-varying partial optimal point $(\hat{x}(t), y^*(t))$, which is a moving target as illustrated in Fig. 4.b). As such, we consider constant step size $\gamma_{n_s} = \mu_{n_f} = \gamma$ instead of diminishing step size.

We need to first quantify the dynamics of the moving partial equilibrium $(\hat{x}(t), y^*(t))$ from the MCTS under the variation of $h^l(t)$. Using the CSI timescale separation property $\epsilon \ll a_H$ for $h^s(t)$ and $h^l(t)$, the dynamics of the moving target $(\hat{x}(t), y^*(t))$ is given by:

Lemma 1 (Dynamics of the moving partial equilibrium): Define $\tilde{G}(x, y, h^s, h^l) = G(x, y, h^s, h^l) +$

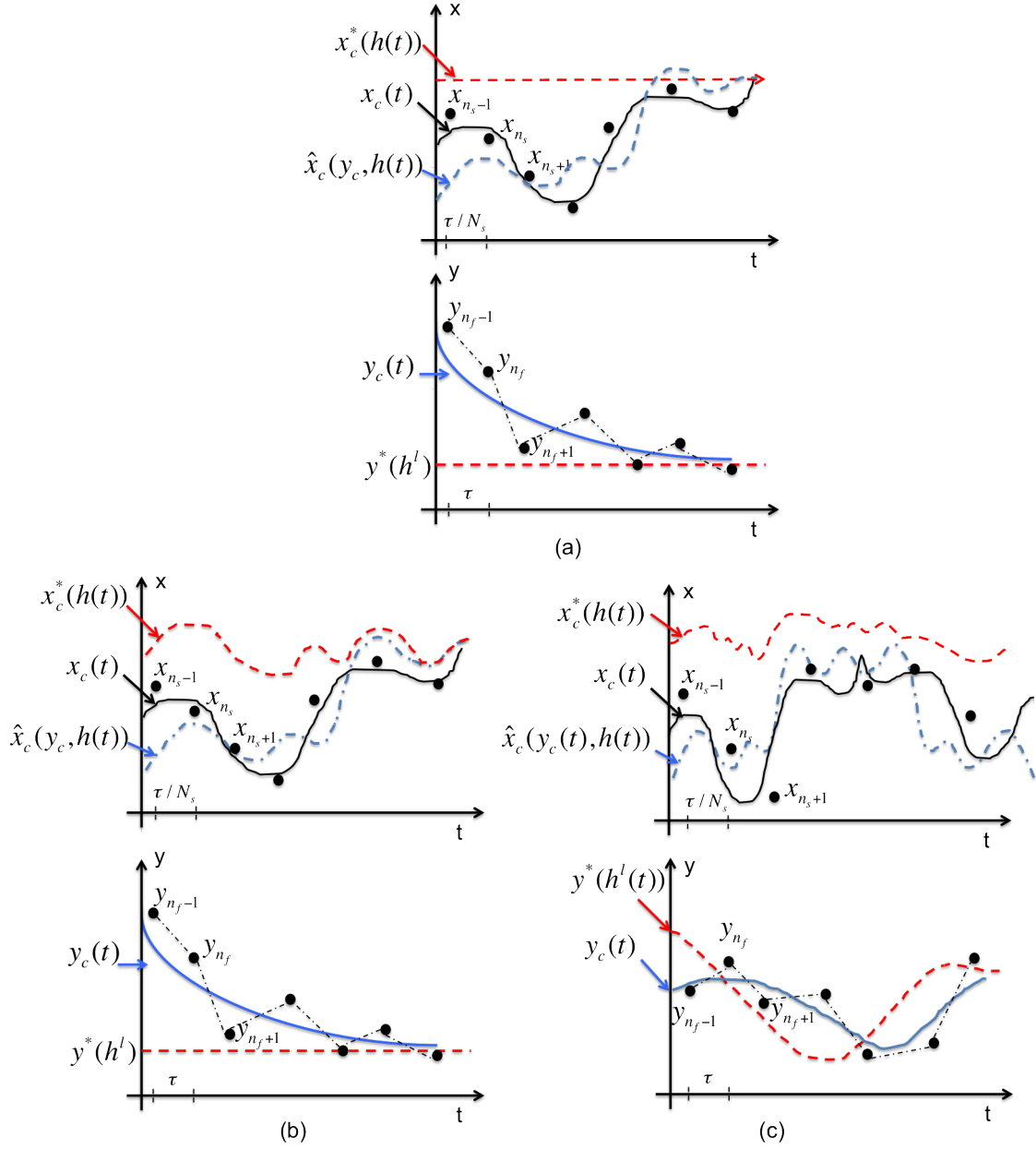


Figure 4. Illustrations of the algorithm iterations, the virtual dynamic system MCTS and the moving equilibrium of the MCTS. (a) illustrates the case for static h^l and h^s , (b) illustrates the case for static h^l and time-varying $h^s(t)$, and (c) illustrates the case of time-varying $h^l(t)$ and $h^s(t)$.

z_x . The dynamics of $\hat{x}_c(t)$ and $y_c^*(t)$ are given by

$$d\hat{x}_c = -\tilde{G}_x^{-1}(\hat{x}_c(y_c, \cdot), y_c, h^s, h^l) \left[\tilde{G}_{h^s}(\hat{x}_c(y_c, \cdot), y_c, h^s, h^l) dh^s + \tilde{G}_y(\hat{x}_c(y_c, \cdot), y_c, h^s, h^l) dy \right] \quad (33)$$

$$dy_c^* = \psi_{h^l}(h^l) dh^l \quad (34)$$

where $\tilde{G}_x(\cdot) \triangleq \frac{\partial}{\partial x} \tilde{G}(x_c, y_c, h^s, h^l)$, $\tilde{G}_y(\cdot) \triangleq \frac{\partial}{\partial y} \tilde{G}(x_c, y_c, h^s, h^l)$, $\tilde{G}_{h^s}(\cdot) \triangleq \frac{\partial}{\partial h^s} \tilde{G}(x_c, y_c, h^s, h^l)$ and $\psi_{h^l} = \frac{\partial}{\partial h^l} \psi(h^l)$ as defined in Assumption 4. ■

Proof: Please refer to Appendix C for the proof. ■

With the notion of MCTS and the dynamic of the moving equilibrium in (33)-(34), the instantaneous tracking errors can be expressed as

$$\begin{aligned} x_{n_s(t)} - \hat{x}_c(t) &= (x_{n_s(t)} - x_c(t)) + (x_c(t) - \hat{x}_c(t)) \\ &= \sqrt{\gamma} \tilde{x}_c^\gamma(t) + \tilde{x}_c^e(t) \end{aligned} \quad (35)$$

and

$$\begin{aligned} y_{n_f(t)} - y_c^*(t) &= (y_{n_f(t)} - y_c(t)) + (y_c(t) - y_c^*(t)) \\ &= \sqrt{\gamma} \tilde{y}_c^\gamma(t) + \tilde{y}_c^e(t) \end{aligned} \quad (36)$$

where $\tilde{x}_c^\gamma(t) \triangleq \frac{1}{\sqrt{\gamma}} (x_{n_s(t)} - x_c(t))$ and $\tilde{y}_c^\gamma(t) \triangleq \frac{1}{\sqrt{\gamma}} (y_{n_f(t)} - y_c(t))$ are the scaled error gap between the iteration trajectory (21)-(22) and the MCTS, and $\tilde{x}_c^e(t) \triangleq x_c(t) - \hat{x}_c(t)$ and $\tilde{y}_c^e(t) \triangleq y_c(t) - y_c^*(t)$ are the scaled tracking error from the MCTS to the moving partial equilibrium (33)-(34).

Even though it is very hard to quantify the tracking errors $x_{n_s(t)} - \hat{x}_c(t)$ and $y_{n_f(t)} - y_c^*(t)$, we can try to study the distributions of the decomposed error states $\tilde{x}_c^\gamma(t)$, $\tilde{x}_c^e(t)$, $\tilde{y}_c^\gamma(t)$, and $\tilde{y}_c^e(t)$ with the help of a virtual dynamic system defined in the following.

Define a joint virtual state $u(t) = (\tilde{x}_c(t), \tilde{y}_c(t), \tilde{x}_c^e(t), \tilde{y}_c^e(t), \tilde{h}^s(t)) \in \mathbb{R}^{2N_x+2N_y+N}$, where $\tilde{x}_c, \tilde{x}_c^e \in \mathbb{R}^{N_x}$, $\tilde{y}_c, \tilde{y}_c^e \in \mathbb{R}^{N_y}$ and $\tilde{h}^s(t)$ is a short-term virtual CSI state with initial value $\tilde{h}^s(0) = h^s(0)$. Correspondingly, define a virtual long-term CSI state $\tilde{h}^l(t)$ as the solution of $d\tilde{h}^l = -\frac{\tau}{N_s\gamma} H_L(t)dt$, with initial value $\tilde{h}^l(0) = h^l(0)$, where $H_L(t)$ is an $N \times N$ diagonal matrix, with the j -th diagonal element being $c_0 \iota D_j(t)^{-\iota-1} v_j(t)$. We use a short hand notation $\tilde{G}_x^{-1} \tilde{G}_{h^s}(\cdot)$ to stand for the matrix $\tilde{G}_x^{-1}(x_c, y_c, h^s, h^l; t) \tilde{G}_{h^s}(x_c, y_c, h^s, h^l; t)$ from (34). The VSDES is defined as follows.

Definition 7 (Virtual stochastic dynamic system (VSDES)): The virtual state trajectory of the VSDES $u(t)$ is characterized by the following SDE:

$$du = U_1(t, u)dt + U_2(t, u)dW_t + dZ_u \quad (37)$$

where

$$U_1(t, u) \triangleq \begin{pmatrix} G_x(x_c, y_c, \tilde{h}^s, \tilde{h}^l) \tilde{x}_c + G_y(x_c, y_c, \tilde{h}^s, \tilde{h}^l) \tilde{y}_c \\ N_s^{-1} K_x(\hat{x}_c, y_c, \cdot) (\tilde{x}_c + \tilde{x}_c^e) + N_s^{-1} K_y(\hat{x}_c, y_c, \cdot) \tilde{y}_c \\ G(x_c, y_c, \tilde{h}^s, \tilde{h}^l) - \frac{1}{2} \frac{a_H \tau}{N_s \gamma} \tilde{G}_x^{-1} \tilde{G}_{h^s}(\cdot) \tilde{h}^s + \tilde{G}_x^{-1} \tilde{G}_y(\cdot) N_s^{-1} k(y_c, \tilde{h}^l) \\ N_s^{-1} k(y_c, \tilde{h}^l) + \frac{\tau}{N_s \gamma} \psi_{h^l}(\tilde{h}^l) H_L(t) \\ - \frac{1}{2} \frac{a_H \tau}{N_s \gamma} \tilde{h}^s \end{pmatrix},$$

and

$$U_2(t, u) \triangleq \begin{pmatrix} \mathbf{0}_{N_x \times N_x} & \dots & \mathbf{0}_{N_x \times N} \\ & \sqrt{\tau N_s^{-1}} \Sigma^{\frac{1}{2}}(y_c; \tilde{h}^l) & \vdots \\ & & \mathbf{0}_{N_x \times N_x} & \sqrt{\frac{a_H \tau}{N_s \gamma}} \tilde{G}_x^{-1} \tilde{G}_{h^s} \\ \vdots & & \mathbf{0}_{N_y \times N_y} & \mathbf{0}_{N_x \times N_x} \\ \mathbf{0} & \dots & & \sqrt{\frac{a_H \tau}{N_s \gamma}} \mathbf{I}_N \end{pmatrix}.$$

Moreover, W_t is a $(2N_x + 2N_y + N)$ -dimensional Brownian motion, $dZ_u = (\mathbf{0}_{N_x}, dZ_x + dZ_y, \tilde{G}_x^{-1} \tilde{G}_y(\cdot) dZ_y, dZ_y, \mathbf{0}_N)$ is the reflection term, and $\Sigma(y_c; \tilde{h}^l)$ is the covariance matrix for the stochastic estimator $K(\cdot)$ in (22). ■

As is summarized in the following theorem, the VSDS provides a weak convergence limit (convergence in distribution, c.f. [19], [36]) to the dynamics of the error gaps $\tilde{x}_c^\gamma(t)$, $\tilde{x}_c^e(t)$, $\tilde{y}_c^\gamma(t)$, and $\tilde{y}_c^e(t)$, when $\gamma \rightarrow 0$.

Theorem 3 (Algorithm Tracking Errors and VSDS): Assuming CSI timescale separation for $h^l(t)$ and $h^s(t)$, i.e., $\epsilon \ll a_H$, the joint state $(\tilde{x}_c^\gamma, \tilde{y}_c^\gamma, \tilde{x}_c^e, \tilde{y}_c^e, \tilde{h}^s)$ weakly converges to $u(t)$, as $\gamma \rightarrow 0$, which is the solution to the VSDS in (37). ■

Proof: Please refer to Appendix D for the proof. ■

The above results allow us to work with the VSDS in (37) to study the convergence of the mixed timescale tracking algorithm (21)-(22). The VSDS provides statistical dynamics for the decomposed tracking error states. We formally summarize the connection between the VSDS and the tracking errors for the mixed-timescale iterations (21)-(22) in the following theorem.

Corollary 2 (Connections between the VSDS and the iterations): Assuming CSI timescale separation for $h^l(t)$ and $h^s(t)$, i.e., $\epsilon \ll a_H$, the tracking errors for the algorithm iterations (21)-(22) defined in (28) and (29) can be upper bounded from $u(t)$, i.e.,

$$e_x \leq \limsup_{t \rightarrow \infty} \frac{1}{t} \max(\gamma, 1) \int_0^t \mathbb{E} [\|\tilde{x}_c(s)\|^2 + \|\tilde{x}_c^e(s)\|^2] ds$$

and

$$e_y \leq \limsup_{t \rightarrow \infty} \frac{1}{t} \max(\gamma, 1) \int_0^t \mathbb{E} [\|\tilde{y}_c(s)\|^2 + \|\tilde{y}_c^e(s)\|^2] ds$$

where $\tilde{x}_c(t)$, $\tilde{x}_c^e(t)$, $\tilde{y}_c(t)$, and $\tilde{y}_c^e(t)$ are the components of the joint state $u(t)$ in the VSDS (37). ■

Corollary 2 can be seen from the tracking errors (35)-(36) and the results in Theorem 3. With Theorem 2, we can focus on studying the *stochastic stability* (formally defined in Section IV-A) of the VSDS in (37) in order to understand the convergence behavior of the mixed timescale iterations in (21) and (22).

Moreover, the VSDS suggests that the tracking errors consist of two parts: (i) the *steady state error* $\gamma \|\tilde{x}_c\|^2$ and $\gamma \|\tilde{y}_c\|^2$, which are the mean square error gaps between the iteration trajectory (x_{n_s}, y_{n_f})

in (21)-(22) and the MCTS $(x_c(t), y_c(t))$ in (30)-(31), and (ii) the *mean tracking error* $\|\tilde{x}_c^e(t)\|^2$ and $\|\tilde{y}_c^e(t)\|^2$, which are the mean square distances between the MCTS and the target moving partial equilibrium $(\hat{x}(t), y^*(t))$ in (33)-(34). Note that when h^l is static, i.e., $H_L(t) \equiv 0$, the tracking error $\tilde{y}_c^e(t)$ in the VSDS converges to 0, and hence, there is only steady state error $\gamma\|\tilde{y}_c\|^2$ (due to constant step size) for the long-term variable y .

IV. CONVERGENCE ANALYSIS OF MIXED TIMESCALE ITERATIONS

In this section, we derive the tracking error bound of the mixed timescale iteration (21)-(22) by studying the VSDS obtained from Section III. We first briefly review the Lyapunov stochastic stability techniques. By studying the stability of the VSDS, we then derive a sufficient condition for the convergence of the mixed-timescale algorithm. Moreover, we obtain a tracking error bound in terms of the parameters of the exogenous process $h(t)$.

A. Preliminary Results on the Lyapunov Stochastic Stability

It is very hard to derive the exact solutions for the VSDS. Instead, we are interested in the expected upper bound value of the joint state $\|u(t)\|$, which represents the aggregated tracking error $e_x + e_y$ of the iterations. This is captured by the *stochastic stability* in mean square defined as follows.

Definition 8 (Stochastic stability in mean square): Given any initial state $u(0) \in \mathcal{U}$, the stochastic process $u(t)$ is globally stochastically stable in mean square, if there exists $0 \leq \delta < \infty$, such that $\limsup_{t \rightarrow \infty} \frac{1}{t} \int_0^t \mathbb{E} \|u(\tau)\|^2 d\tau \leq \delta$. ■

We use a Lyapunov method to study the stochastic stability of $u(t)$. Define a non-negative function $V(u) = \frac{1}{2}u^T u$ along the trajectory of $u(t)$. The function has the property that $V(u) \sim \|u\|^2$, which plays the role of an energy function, where a larger $\|u\|$ gives a larger function value. We summarize the main techniques of stochastic stability analysis as follows.

Definition 9 (Lyapunov drift operator): Consider a stochastic process $u(t)$ and a real-valued Lyapunov function $V(u)$. The Lyapunov drift operator is an infinitesimal estimator on $V(\cdot)$ defined as $\tilde{\mathcal{L}}V(u) = \lim_{\delta \downarrow 0} \frac{1}{\delta} [\mathbb{E} [V(u(t+\delta)|V(u(t)) - V(u(t))]$. ■

Lemma 2 (Stochastic Stability from Lyapunov Drift): Consider a function $f(u)$ that satisfies $f(u) \geq a\|u\|^r$ for all $u \in \mathcal{U}$, and some $a, r > 0$. Suppose the stochastic Lyapunov drift of the process $u(t)$ has the following property

$$\tilde{\mathcal{L}}V(u) \leq -f(u) + g(s) \quad (38)$$

for all $u \in \mathcal{U}$, where $s(t)$ is a stochastic process that satisfies $\limsup_{t \rightarrow \infty} \frac{1}{t} \int_0^t \mathbb{E} [g(s(\tau))] d\tau \leq d$ for some function $g(s)$ and $d < \infty$. Then the process $u(t)$ is stochastically stable, and

$$\limsup_{t \rightarrow \infty} \frac{1}{t} \int_0^t \mathbb{E} \|u(\tau)\|^r d\tau \leq \frac{d}{a}.$$

■

The above result is based on the Foster-Lyapunov criteria for continuous time processes in [40] and is a simple extension of the results in [24, Theorem 2]. The advantage of the Lyapunov method enables a qualitative analysis of the SDE without explicitly solving it. Using such a technique, we derive the stability results for the mixed timescale algorithm in the following.

B. Stability of the Mixed-Timescale Algorithm

Corresponding to stability of a random process, we define the iteration stability as follows.

Definition 10 (Stability of the iteration): The iterations (21) and (22) are stable if the corresponding tracking error defined in (28) and (29) are bounded, i.e., there exists a $B < \infty$, such that $e_x + e_y \leq B$. ■

Note that due to the stochastic iterations, the tracking errors defined in (28) and (29) may be unbounded statistically. To study the algorithm stability, we can equivalently investigate the stability of the VSDS $u(t)$. Towards this end, we first construct a Lyapunov drift $\tilde{\mathcal{L}}V(u)$ on the trajectory of the VSDS in (37). We then proceed to find a function $g(s)$ that satisfies condition (38). Finally, we use Theorem 2 to obtain the stability result.

We summarize the sufficient conditions of the stability of the mixed timescale algorithm as follows.

Theorem 4 (Sufficient conditions for the algorithm stability): Assume CSI timescale separation for $h^l(t)$ and $h^s(t)$, i.e., $\epsilon \ll a_H$. Suppose that there exist $0 < v_H, v_y < \infty$, such that $\|G_x^{-1}G_{h^s}\| \leq v_H$ and $\|G_x^{-1}G_y\| \leq v_y$. Then the sufficient condition for the algorithm to be stable is given by

$$\alpha N_s \left(8\alpha_x - \frac{a_H \tau}{N_s \gamma} v_H^2 \right) - 2l_x^2 - 2l_y^2 v_y^2 > 0. \quad (39)$$

■

Proof: Please refer to Appendix E for the proof. ■

The above results have several implications on the convergence.

- *Convergence of the inner problem:* The term $N_s \left(8\alpha_x - \frac{a_H \tau}{N_s \gamma} v_H^2 \right)$ specifies the convergence behavior of the inner problem. Recall that the parameter a_H controls the variation speed of the fast changing CSI $h^s(t)$, α_x represents the convergence rate of the inner problem, τ is the frame duration, N_s is the number of slots per frame, and γ is the step size. As such, for a given a_H , we need to have sufficiently fast inner iterations (α_x) or sufficient number of slots per frame (N_s) in order to have bounded tracking error.
- *Convergence of the outer problem and the coupling effect:* The convergence of the inner problem and outer problem is coupled together. The stability of the inner problem (a positive $8\alpha_x - \frac{a_H \tau}{N_s \gamma} v_H^2$) is a premise of the stability of the whole algorithm. To achieve the stability, we desire small

v_H and l_x , which represent the sensitivity of the partial stationary point \hat{x} w.r.t. the change of $h^s(t)$. On the other hand, we also want the parameters l_y and v_y to be small, which means that y^* shall not be quite sensitive to the bias induced by the tracking error $x_{n_s} - \hat{x}(y_{n_f}, \cdot)$.

- *Impact from the iteration timescale:* One can reduce the frame duration τ , increase the number of slots N_s per frame, or increase the step size γ to enhance the stability of the overall algorithm. However, shortening the frame duration τ may result in a larger amount of signaling overhead to update the long-term variable y_{n_f} and the acquisition of local CSI $h^s(n_f\tau)$; increasing the number of slots N_s yields a higher computational complexity; and moreover, a large step size γ may give larger steady state error $\mathcal{O}(\gamma)$ for the discrete-time trajectory.

C. Upper Bound of the Tracking Error

Stability is only a weak result of convergence. We are interested in the tracking error bound of the algorithm. Under the sufficient condition specified in (39), using the Lyapunov technique in Lemma 2, we study the result on the upper bound of the tracking errors e_x and e_y .

Theorem 5 (Upper bound of the tracking error): Assume the conditions in Theorem 4. If $\bar{\Sigma} \triangleq \limsup_{t \rightarrow \infty} \frac{1}{t} \int_0^t \text{tr}(\Sigma(y^*(h^l(\tau)))) d\tau < \infty$, the tracking errors e_x and e_y are given by:

$$e_x + e_y \leq \frac{\eta}{\rho} (\tau \bar{\Sigma} + C) \quad (40)$$

where $\rho = \mathcal{O}(N_s \alpha^2 \alpha_x \alpha_y)$, $\eta = \mathcal{O}(\sqrt{N_s^2 \alpha_x^2 + \alpha^2})$, $C = \frac{a_H \tau}{\gamma} N(1 + v_H^2) + \mathcal{O}(\epsilon^2 \varpi^2 \tau^2 \gamma^{-2})$, and N is the dimension of the CSI vector h^s . ■

Proof: Please refer to Appendix F for the proof. ■

The above result shows that the upper bound of the tracking error depends on several important parameters, such as the timescale parameter a_H for $h^s(t)$, the sensitivities v_H and v_y of the stationary points \hat{x} and y^* , respectively, as well as the sensitivities ϖ of $y^*(h^l)$ over h^l . A faster time-varying scenario corresponds to larger a_H , which result in a larger tracking error bound. In addition, we can observe the followings.

- *Special case for static h^l and h^s :* Under static CSI, where the CSI timescale parameters $a_H = \epsilon = 0$, we have the term $C = 0$ in the error bound. The tracking error is governed by the steady state error $\frac{\eta}{\rho} \tau \bar{\Sigma}$ due to the constant step size γ . Note that if diminishing step size is used for the outer iteration (22), i.e., $\mu_{n_f} \rightarrow 0$, the tracking error bound in (40) becomes 0. This corresponds to the case in Fig. 4.a).
- *Special case for static h^l and time-varying $h^s(t)$:* Under static h^l , where long-term CSI timescale parameter $\epsilon = 0$, the term C decreases, because the term $\mathcal{O}(\epsilon^2 \varpi^2 \tau^2 \gamma^{-2})$ becomes 0. In particular, if diminishing step size is used for the outer iteration (22), i.e., $\mu_{n_f} \rightarrow 0$, the error bound becomes

$\frac{\eta}{\rho}C$ and is mainly contributed by the tracking error in the inner iteration. This corresponds to the case in Fig. 4.b).

- *Impact of the algorithm parameters:* When the CSI (h^s, h^l) is fast changing, i.e., the CSI timescale parameters a_H and ϵ are large, one can reduce the frame duration τ , increase the number of slots N_s per frame, or increase the step size γ to reduce the tracking error, with the price of larger signaling overhead, higher computational complexity and larger steady state error $\mathcal{O}(\gamma)$ as discussed in Section IV-B.

V. COMPENSATION FOR MIXED TIMESCALE ITERATIONS

In the previous sections, we have analyzed the convergence behavior of the iteration (21)-(22) under mixed timescale time-varying CSI $h(t)$. In this section, we shall enhance the algorithm for better tracking performance. Since the convergence of the outer long-term variable y depends on the convergence of the inner problem, it is essential to accelerate the convergence of the short-term variable x_{n_s} . Towards this end, we introduce a compensation term in the algorithm (21) to offset the exogenous disturbance to the VSDS in (37).

A. Adaptive Compensations for the Time-varying CSI

Recall that in the mixed timescale iterations (21)-(22), the inner iteration tracks the moving target $\hat{x}(t)$ driven by the time-varying $h(t)$ and $y(t)$. On the other hand, the outer iteration tracks the moving target $y^*(t)$ driven by $h^l(t)$. As a result, when $h(t)$ and $y(t)$ are time-varying, they generate disturbance to the tracking iterations (21) and (22). This can be seen from the dynamics of the error states $\tilde{x}_c^e(t)$ and $\tilde{y}_c^e(t)$ in the VSDS in (37) (c.f. equations (75) and (76))

$$d\tilde{x}_c^e = G(x_c, y_c, \tilde{h}^s, \tilde{h}^l)dt + \underbrace{\tilde{G}_x^{-1}\tilde{G}_{h^s}(\cdot)d\tilde{h}^s}_{\text{exogenous disturbance from } \tilde{h}^s(t)} + \underbrace{\tilde{G}_x^{-1}\tilde{G}_y(\cdot)dy_c}_{\text{disturbance from } y(t)} \quad (41)$$

$$d\tilde{y}_c^e = N_s^{-1}k(y_c, \tilde{h}^l)dt + \underbrace{-\psi_{h^l}(h^l)d\tilde{h}^l}_{\text{exogenous disturbance from } \tilde{h}^l(t)}. \quad (42)$$

Note that, from Theorem 2 and Corollary 1, when $d\tilde{h}^l = 0$, the error dynamic system (42) is asymptotically stable and the error state $\tilde{y}_c^e(t)$ converges to 0. In addition, when $d\tilde{h}^s = 0$, the system (41) is stable and the error state $\tilde{x}_c^e(t)$ converges to 0 as well. However, with the presence of the *exogenous disturbance* $d\tilde{h}^s$ and $d\tilde{h}^l$, the error states $\tilde{x}_c^e(t)$ and $\tilde{y}_c^e(t)$ are continuously disturbed and may fail to converge to the origin. This is illustrated in Fig. 5.

Such observation motivates us to offset the exogenous disturbance to reduce the tracking error under time-varying $h^s(t)$ and $h^l(t)$. We start by introducing compensation terms to the MCTS (30)-(31) as

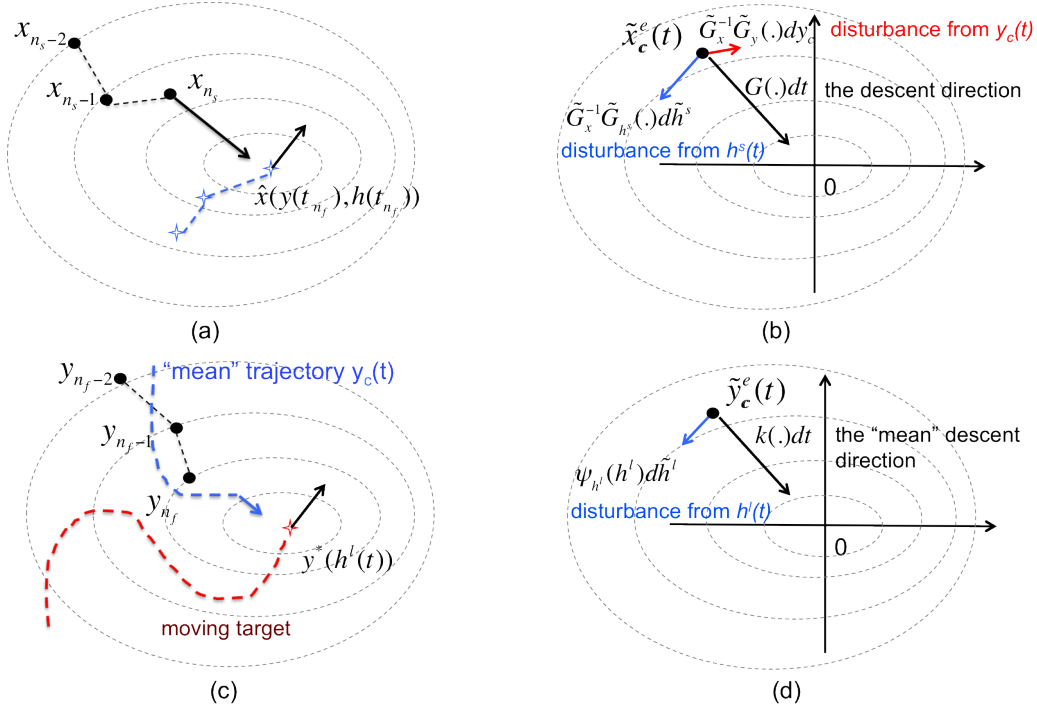


Figure 5. Illustrations of the convergence of the iterations (x_{n_s}, y_{n_f}) and the virtual error dynamic systems $\tilde{x}_c^e(t)$ and $\tilde{y}_c^e(t)$. (a) The algorithm iteration x_{n_s} tracks the target $\hat{x}(y(t), h(t))$, which is moving due to the time-varying $y(t)$ and $h(t)$. (b) The corresponding error state $\tilde{x}_c^e(t)$ approaches to 0 under the mapping $G(\cdot)$. However, the exogenous disturbance $\tilde{G}_x^{-1}\tilde{G}_{h^s}(\cdot)d\tilde{h}^s$ and $\tilde{G}_x^{-1}\tilde{G}_y(\cdot)dy_c$ drag it away from the origin. (c) The algorithm iteration y_{n_f} tracks $y^*(h^l(t))$ following the “mean” virtual trajectory $y_c(t)$. (d) The corresponding error state $\tilde{y}_c^e(t)$ approaches to 0 following the virtual direction $k(\cdot)$. However, the exogenous disturbance $\psi_{h^l}(\tilde{h}^l)d\tilde{h}^l$ drags it away from the origin.

follows,

$$dx_c = G(x_c, y_c, \tilde{h}^s, \tilde{h}^l)dt \underbrace{-\widehat{\tilde{G}_x^{-1}\tilde{G}_{h^s}(\cdot)}d\tilde{h}^s}_{\text{compensation for } \tilde{h}^s(t)} \underbrace{-\widehat{\tilde{G}_x^{-1}\tilde{G}_y(\cdot)}dy_c}_{\text{compensation for } y(t)} \quad (43)$$

$$dy_c = N_s^{-1}k(y_c, \tilde{h}^l)dt \underbrace{+\widehat{\psi_{h^l}(\tilde{h}^l)}d\tilde{h}^l}_{\text{compensation for } \tilde{h}^l(t)} \quad (44)$$

where $-\widehat{\tilde{G}_x^{-1}\tilde{G}_{h^s}(\cdot)}d\tilde{h}^s$ and $-\widehat{\tilde{G}_x^{-1}\tilde{G}_y(\cdot)}dy_c$ are compensation terms to offset the disturbance in (41), and $\widehat{\psi_{h^l}(\tilde{h}^l)}d\tilde{h}^l$ is a compensation term to offset the disturbance in (42). Here, for a simple discussion, we drop the reflection terms dZ_x and dZ_y , since the projections that drive the reflections are always

conservative⁵. As a result, the dynamic system in (41)-(42) becomes

$$\begin{aligned} d\tilde{x}_c^e &= G(x_c, y_c, \tilde{h}^s, \tilde{h}^l)dt + \left(\tilde{G}_x^{-1}\tilde{G}_{h^s}(\cdot) - \widehat{\tilde{G}_x^{-1}\tilde{G}_{h^s}(\cdot)} \right) d\tilde{h}^s \\ &\quad + \left(\tilde{G}_x^{-1}\tilde{G}_y(\cdot) - \widehat{\tilde{G}_x^{-1}\tilde{G}_y(\cdot)} \right) dy_c \end{aligned} \quad (45)$$

$$d\tilde{y}_c^e = N_s^{-1}k(y_c, \tilde{h}^l)dt + \left(\widehat{\psi_{h^l}(\tilde{h}^l)} - \psi_{h^l}(\tilde{h}^l) \right) d\tilde{h}^l. \quad (46)$$

Define the disturbance components as $\varphi_x^h(\cdot) \triangleq -\tilde{G}_x^{-1}\tilde{G}_{h^s}(\cdot)$, $\varphi_x^y(\cdot) \triangleq -\tilde{G}_x^{-1}\tilde{G}_y(\cdot)$ and $\varphi_y^h(\cdot) \triangleq \psi_{h^l}(\cdot)$. Consider $\hat{\varphi}_x^h(x; \cdot)$, $\hat{\varphi}_x^y(x; \cdot)$ and $\hat{\varphi}_y^h(y; \cdot)$ as the *compensation estimators* for the disturbance components $\varphi_x^h(\cdot)$, $\varphi_x^y(\cdot)$ and $\varphi_y^h(\cdot)$. The corresponding compensated mixed timescale algorithm is given by,

$$\begin{aligned} x_{n_s} &= \mathcal{P}_X \left[x_{n_s-1} + \gamma G(x_{n_s-1}, y_{n_f}; h^s(n_f\tau), h^l(n_f\tau)) \right. \\ &\quad \left. + \hat{\varphi}_x^h(x_{n_s-1}; \cdot)(\Delta h^s)_{n_s} + \hat{\varphi}_x^y(x_{n_s-1}; \cdot)(\Delta y)_{n_s} \right] \end{aligned} \quad (47)$$

$$y_{n_f} = \mathcal{P}_Y \left[y_{n_f-1} + \gamma K(x_{n_s}, y_{n_f-1}; h^s(n_f\tau), h^l(n_f\tau)) + \hat{\varphi}_y^h(y_{n_f-1}; \cdot)(\Delta h^l)_{n_f} \right] \quad (48)$$

where $(\Delta h^s)_{n_s} = h^s(\lfloor \frac{n_s}{N_s} \rfloor \tau) - h^s(\lfloor \frac{n_s-1}{N_s} \rfloor \tau)$, $(\Delta y)_{n_s} = y_{\lfloor \frac{n_s}{N_s} \rfloor} - y_{\lfloor \frac{n_s-1}{N_s} \rfloor}$, and $(\Delta h^l)_{n_f} = h^l(n_f\tau) - h^l(n_f\tau - \tau)$. The compensation terms are non-zero on the frame boundary.

B. Derivation of the Compensation Estimators

Obviously, if we can precisely estimate the disturbance, its impact to the convergence can be totally suppressed and the tracking errors \tilde{x}_c^e and \tilde{y}_c^e go to zero. Unfortunately, we cannot obtain perfect estimations of the disturbance terms $-\tilde{G}_x^{-1}\tilde{G}_{h^s}(\cdot)$, $-\tilde{G}_x^{-1}\tilde{G}_y(\cdot)$, and $\psi_{h^l}(h^l)$, because they require closed form expressions of the target $(\hat{x}(y, h^s, h^l), y^*(h^l))$. In this section, we derive approximate compensation terms using Lagrange duality theory [32], [33] and implicit function theorem in calculus.

1) *Compensation for the Short-term Iteration x_{n_s}* : Consider the optimality condition [33] for the inner problem (7),

$$\mathcal{G}(\theta_x, \lambda_x; \theta_y, h^s, h^l) = \begin{bmatrix} \nabla_x F(\theta_x, \theta_y; h^s, h^l) - \sum_i \lambda_{x,i} \nabla w_i(\theta_x, \theta_y; \cdot) \\ \{\lambda_{x,i} w_i(\theta_x, \theta_y; \cdot)\}_{i=1}^W \end{bmatrix} = 0$$

where $\lambda_{x,i} \geq 0$ and $w_i(\theta_x, \theta_y; \cdot) \leq 0$, for all $1 \leq i \leq W$. From the Lagrangian duality theory, $\mathcal{G}(x; \theta_y, h^s, h^l) = 0$ has a unique solution $\hat{x} = (\hat{\theta}_x, \hat{\lambda}_x)$ for $\hat{\lambda}_x \geq 0$, and the dynamics of $\hat{x}(t)$ should satisfy $\mathcal{G}_x(\hat{x}; \cdot) \frac{d\hat{x}}{dt} + \mathcal{G}_{h^s}(\hat{x}; \cdot) \frac{dh^s}{dt} + \mathcal{G}_y(\hat{x}; \cdot) \frac{dy_c}{dt} = 0$. Suppose the matrix $\mathcal{G}_x(x; \cdot)$ is invertible. Using the implicit function theorem, the compensation estimators can be given by

$$\hat{\varphi}_x^h(x; \cdot) = -\mathcal{G}_x^{-1}\mathcal{G}_{h^s}(x; \cdot), \text{ and } \hat{\varphi}_x^y(x; \cdot) = -\mathcal{G}_x^{-1}\mathcal{G}_y(x; \cdot). \quad (49)$$

⁵The reflection is from the constraints that form the convex domain $\mathcal{X}(y) \times \mathcal{Y}$. As the constraint domain is to restrict the iteration trajectory, it always helps the convergence.

2) *Compensation for the Long-term Iteration y_n* : From the Lagrange duality theory, the optimality condition for the outer problem (10) is given by

$$\mathcal{T}(\theta_y, \lambda_y; h^l) = \begin{bmatrix} \nabla_y \mathbb{E}F(\theta_x, \theta_y; h^s, h^l) - \sum_j \lambda_{y,j} \nabla q_j(\theta_y; h^l) \\ \{\lambda_{y,j} q_j(\theta_y; h^l)\}_{j=1}^J \end{bmatrix} = 0 \quad (50)$$

where $\lambda_{y,j} \geq 0$ and $q_j(\theta_y; \cdot) \leq 0$, for all $1 \leq j \leq J$. Similarly, $\mathcal{T}(y; \cdot) = 0$ has a unique solution $y^* = (\theta_y^*, \lambda_y^*)$ for $\lambda_y^* \geq 0$, and the dynamics of $y^*(t)$ should satisfy $\mathcal{T}_y(y^*; \cdot) \frac{dy^*}{dt} + \mathcal{T}_{h^l}(y^*; \cdot) \frac{dh^l}{dt} = 0$. Also, using implicit function theorem, an ideal compensation estimator can be given by $-\mathcal{T}_y^{-1} \mathcal{T}_{h^l}(y; \cdot)$.

Note that closed form expression is usually not available for $\mathcal{T}(y; \cdot)$, due to the expectation $\mathbb{E}F(\cdot)$. Alternatively, define $\hat{\mathcal{T}}(\theta_y, \lambda_y; h^s, h^l) \triangleq \begin{bmatrix} \nabla_y F(\theta_x, \theta_y; h^s, h^l) - \sum_j \lambda_{y,j} \nabla q_j(\theta_y; \cdot) \\ \{\lambda_{y,j} q_j(\theta_y; \cdot)\}_{j=1}^J \end{bmatrix}$. Then for given h^l , $\hat{\mathcal{T}}_y(y; \cdot)$ is an unbiased estimator of $\mathcal{T}_y(y; \cdot)$, since $\mathbb{E} \hat{\mathcal{T}}(y; \cdot) = \mathcal{T}(y; \cdot)$. Similarly, $\hat{\mathcal{T}}_{h^l}(y; \cdot)$ is a unbiased estimator of $\mathcal{T}_{h^l}(y; \cdot)$. As a result, the compensation estimator for the long-term iteration can be given by

$$\hat{\varphi}_y^h(y; \cdot) = -\hat{\mathcal{T}}_y^{-1} \hat{\mathcal{T}}_{h^l}(y; \cdot). \quad (51)$$

C. Performance of the Compensation Algorithm

Although the compensation estimators derived in (49) and (51) are from approximation, we can show that under some technical conditions, the tracking errors \tilde{x}_c^e and \tilde{y}_c^e from continuous-time trajectories $x_c(t)$ and $y_c(t)$ can be significantly eliminated.

Theorem 6 (Tracking performance of the compensation algorithm): Assume CSI timescale separation for $h^l(t)$ and $h^s(t)$, i.e., $\epsilon \ll a_H$. Suppose that $\hat{\varphi}_x^h(\cdot)$, $\hat{\varphi}_x^y(\cdot)$ and $\hat{\varphi}_y^h(y; \cdot)$ take the forms in (49) and (51), and they are Lipschitz continuous, i.e., there exist positive constants $\hat{L}_x^h, \hat{L}_x^y, \hat{L}_y^h, \beta_y^h < \infty$, such that $\|\hat{\varphi}^h(x; \cdot) - \hat{\varphi}^h(\hat{x}; \cdot)\| \leq \hat{L}_x^h \|x - \hat{x}\|$, $\|\hat{\varphi}^y(x; \cdot) - \hat{\varphi}^y(\hat{x}; \cdot)\| \leq \hat{L}_x^y \|x - \hat{x}\|$ and $\mathbb{E} \|\hat{\varphi}_y^h(y; \cdot) - \varphi_y^h(\cdot)\| \leq \hat{L}_y^h \|y - y^*\| + \beta_y^h$, for all $x \in \mathcal{X}(y)$, $y \in \mathcal{Y}$. Then, if

$$\begin{aligned} \text{(i)} \quad & \alpha_y \gamma - \epsilon \tau \hat{L}_y^h > 0, \text{ and} \\ \text{(ii)} \quad & \alpha_x - \frac{a_H \tau}{\sqrt{2\pi} N_s \gamma} \hat{L}_x^h - \hat{L}_x^y N_s^{-1} l_y \frac{\epsilon \tau \beta_y^h}{\alpha_y \gamma - \epsilon \tau \hat{L}_y^h} - \frac{a_H \tau}{N_s \gamma} \left(\hat{L}_x^h \right)^2 > 0, \end{aligned}$$

the tracking error \tilde{x}_c^e converges to 0 in probability, and the tracking error for \tilde{y}_c^e is upper bounded by $\mathbb{E} \|\tilde{y}_c^e\| \leq \frac{\epsilon \tau \beta_y^h}{\alpha_y \gamma - \epsilon \tau \hat{L}_y^h}$. ■

Proof: Please refer to Appendix G for the proof. ■

In the case when h^l is static, the convergence result is given in the following corollary.

Corollary 3 (Tracking performance under static h^l): Suppose $\hat{\varphi}_x^h(\cdot)$ and $\hat{\varphi}_x^y(\cdot)$ are given by (49) and Lipschitz continuous as specified in Theorem 6. Then if $\alpha_x - \frac{a_H \tau}{\sqrt{2\pi} N_s \gamma} \hat{L}_x^h - \hat{L}_x^y N_s^{-1} l_y \frac{\epsilon \tau \beta_y^h}{\alpha_y \gamma - \epsilon \tau \hat{L}_y^h} - \frac{a_H \tau}{N_s \gamma} \left(\hat{L}_x^h \right)^2 > 0$, the tracking error \tilde{x}_c^e converges to 0 in probability. ■

Corollary 3 is obtained by setting the timescale parameter $\epsilon = 0$ in Theorem 6. It corresponds to the case 1 scenario studied in Section III-A. However, the performance with compensation is stronger because it does not require the short-term CSI timescale to be extremely slower than the algorithm timescale (i.e., $a_H\tau \ll \gamma$) in Theorem 2 and Corollary 1 for the convergence.

Remark 4 (Interpretation of the results): Theorem 6 and Corollary 3 show the performance advantage of the compensation algorithm under time-varying CSI. Specifically, we have the following observations.

- *Compensation for the long-term control:* When the bias β_y^h of the compensation estimator goes to 0, the tracking error $\tilde{y}_c^e = y_c - y^*$ of the long-term variable y converges to 0 as well. Note that the bias β_y^h comes from using $\nabla_y F(\cdot)$ to estimate $\nabla_y \mathbb{E}F(\cdot)$ in (50). To reduce the bias, we can use a Monte-Carlo method to estimate $\widehat{\nabla_y \mathbb{E}F(\cdot)} = \frac{1}{M} \sum_{m=1}^M \nabla_y F(x_m, y_0)$, by observing many realizations of $\nabla_y F(x(t), y(t_0); \cdot)$ in the inner timescale.
- *Compensation for the short-term control:* Theorem 6 implies that when the short timescale CSI $h^s(t)$ does not change too fast (moderate a_H), the compensation algorithm can keep track with the stationary point target. This is a much weaker condition for the conventional convergence result, which requires $a_H\tau \ll \gamma$, i.e., the algorithm must iterate much faster than the CSI dynamics. Moreover, one can reduce the frame duration τ , increase the number of slots N_s per frame, or increase the step size γ to satisfy condition (ii) for enhancing the tracking of the inner iteration x_{n_s} , at the cost of larger signaling overhead, higher computational complexity and larger steady state error $\mathcal{O}(\gamma)$ as discussed in Section IV-B.

■

Note that, even though there can be zero convergence errors for x_c and y_c in (43) and (44), the discrete-time iterations (47) and (48) still have $\mathcal{O}(\gamma)$ steady state error due to the constant step size γ used for the tracking. Nevertheless, Theorem 6 and Corollary 3 indicate that the proposed compensation algorithm has an eminent convergence capability under time-varying $h^s(t)$ and $h^l(t)$.

VI. AN APPLICATION EXAMPLE: RESOURCE ALLOCATIONS IN WIRELESS MULTI-HOP RELAY NETWORK

The mixed timescale optimization approach has vast applications in wireless communication networks. In the following, we consider a particular example of joint flow control and power allocation in wireless relay network described in Section II-B. From this example, we demonstrate the compensation algorithm and apply the theoretical results for the convergence analysis.

A. The Two-Timescale Algorithm

We apply the stochastic primal-dual method in (26) to derive the iterative algorithm x_{n_s} and y_{n_f} in this example. Denote $x = (\mathbf{p}, \lambda)$, where $\lambda = (\lambda_1, \dots, \lambda_W)$ is the Lagrange multiplier. For the example problem in (13), we can form the Lagrange function as

$$L(\mathbf{p}, \lambda, \mathbf{r}; \mathbf{h}) = \sum_{j \in \mathcal{E}} \log(r_j) - V \sum_{k \in \mathcal{L}} p_k - \sum_{j=1}^J \lambda_j \left[\sum_{k \in \mathcal{S}_j} c_k(\mathbf{r}) - \log \left(1 + \sum_{k \in \mathcal{S}_j} |h_k|^2 p_k \right) \right] \quad (52)$$

where W is the total number of constraints, and $\mathcal{S}_j \subset \mathcal{L}^+(m)$ for some $m \in \mathcal{R} \cup \{0\}$.

1) *Iteration for the short-term variable:* The optimality condition (KKT condition [41]) for the inner problem is given by

$$\mathcal{G}(\mathbf{p}, \lambda; \mathbf{r}, \mathbf{h}) = \begin{bmatrix} \frac{\partial}{\partial \mathbf{p}} L(\mathbf{p}, \lambda, \mathbf{r}) \\ \left\{ \lambda_j \left[\sum_{k \in \mathcal{S}_j} c_k(\mathbf{r}) - \log \left(1 + \sum_{k \in \mathcal{S}_j} |h_k|^2 p_k \right) \right] \right\}_{j=1}^W \end{bmatrix} = \mathbf{0}.$$

Following the adaptive compensation algorithm in Section V, the iteration of the short-term variable is given by

$$\mathbf{p}(n_s + 1) = \mathcal{P}_{\mathbf{p}} \left[\mathbf{p}(n_s) + \gamma \frac{\partial}{\partial \mathbf{p}} L(\mathbf{p}(n_s), \lambda(n_s); \mathbf{r}(n_f)) + \hat{\Psi}_p(\cdot) \right] \quad (53)$$

$$\lambda(n_s + 1) = \mathcal{P}_{\lambda} \left[\lambda(n_s) - \gamma \frac{\partial}{\partial \lambda} L(\mathbf{p}(n_s), \lambda(n_s); \mathbf{r}(n_f)) + \hat{\Psi}_{\lambda}(\cdot) \right] \quad (54)$$

where the projection $\mathcal{P}_{\mathbf{p}}(\cdot)$ and $\mathcal{P}_{\lambda}(\cdot)$ are to restrict the elements to be non-negative. The term $(\frac{\partial}{\partial \mathbf{p}} L(\cdot), \frac{\partial}{\partial \lambda} L(\cdot))$ corresponds to the iteration mapping $G(\cdot)$ in (21) (and (47)). The compensations $\hat{\Psi}_p(\cdot)$ and $\hat{\Psi}_{\lambda}(\cdot)$ can be derived as

$$\begin{pmatrix} \hat{\Psi}_p(\cdot) \\ \hat{\Psi}_{\lambda}(\cdot) \end{pmatrix} = -\mathcal{G}_{(\mathbf{p}, \lambda)}^{-1} \mathcal{G}_{\mathbf{h}}(\mathbf{p}(n_s), \cdot) \Delta \mathbf{h}(n_s) - \mathcal{G}_{(\mathbf{p}, \lambda)}^{-1} \mathcal{G}_{\mathbf{y}}(\mathbf{p}(n_s), \cdot) \Delta \mathbf{r}(n_s)$$

where $\Delta \mathbf{h}(n_s) = \mathbf{h}(\lfloor \frac{n_s}{N_s} \rfloor \tau) - \mathbf{h}(\lfloor \frac{n_s-1}{N_s} \rfloor \tau)$ and $\Delta \mathbf{r}(n_s) = \mathbf{r}(\lfloor \frac{n_s}{N_s} \rfloor) - \mathbf{r}(\lfloor \frac{n_s-1}{N_s} \rfloor)$.

2) *Iteration for the long-term variable:* We first derive an augmented Lagrange function $L_1(\cdot)$ by substituting the equality constraints (20) into the Lagrangian $L(\cdot)$ in (52). The optimality condition for the outer problem is given by

$$\mathcal{T}(\mathbf{r}; \mathbf{h}^l) = \frac{\partial}{\partial \mathbf{r}} \mathbb{E} L_1(\mathbf{p}(n_s), \lambda(n_s), \mathbf{r}(n_f)) = 0.$$

The update of the long-term variable is given by

$$\mathbf{r}(n_f + 1) = \mathcal{P}_{\mathbf{r}} \left[\mathbf{r}(n_f) + \gamma \frac{\partial}{\partial \mathbf{r}} L_1(\mathbf{p}(n_s), \lambda(n_s), \mathbf{r}(n_f)) + \hat{\Psi}_r(\cdot) \right] \quad (55)$$

where the projection $\mathcal{P}_{\mathbf{r}}(\cdot)$ is to restrict \mathbf{r} to be non-negative. The iteration (55) corresponds to the long-term variable update for y_{n_f} in (22), and the term $\frac{\partial}{\partial \mathbf{r}} L_1(\mathbf{p}(n_s); \lambda(n_s), \mathbf{r}(n_f))$ corresponds to the stochastic estimator $K(\cdot)$ in (22). The compensation $\hat{\Psi}_r(\cdot)$ can be derived as

$$\hat{\Psi}_r(\mathbf{r}(n_f), \mathbf{h}^l(n_f \tau)) = -\hat{\mathcal{T}}_r^{-1} \hat{\mathcal{T}}_{h^l}(\mathbf{r}(n_f); \mathbf{h}^l(n_f))(\mathbf{h}^l(n_f \tau) - \mathbf{h}^l((n_f - 1)\tau))$$

Components in the example		Corresponding components in the model	
$(\mathbf{p}(n_s), \lambda(n_s))$	Primal-dual inner iteration (53)-(54)	x_{n_s}	Short-term iterative sequence (21)
$\mathbf{r}(n_f)$	Primal outer iteration (55)	y_{n_f}	Long-term iterative sequence (22)
$(\hat{\Psi}_p(\cdot), \hat{\Psi}_\lambda(\cdot))$	Compensation for the inner iteration	$-\hat{\varphi}_x^h dh^s - \hat{\varphi}_x^y dy$	Compensation for the inner iteration
$\hat{\Psi}_r$	Compensation for the outer iteration	$\hat{\varphi}_y^h dh^l$	Compensation for the outer iteration
$\mathbb{R}_+^{ \mathcal{L} } \times \mathbb{R}_+^W$	Projection domain for the inner iteration	$\mathcal{X}(y)$	Projection domain for x_{n_s}
$\mathbb{R}_+^{ \mathcal{L} }$	Projection domain for the outer iteration	\mathcal{Y}	Projection domain for y_{n_f}

Table II

ALGORITHM ASSOCIATIONS BETWEEN THE EXAMPLE AND THE MIXED TIMESCALE SYSTEM MODEL.

where $\hat{\mathcal{T}}(\mathbf{r}; \mathbf{h}^l) = \frac{\partial}{\partial \mathbf{r}} L_1(\mathbf{p}, \lambda, \mathbf{r}; \mathbf{h}^l)$ and the path loss variable $\mathbf{h}^l(t)$ can be measured by averaging the CSI $\mathbf{h}(t)$ over a certain time window.

Table II summaries the algorithm association between the example and the mixed timescale model.

B. Implementation Considerations

With the iteration timescale decomposition, we can consider two implementation scenarios of the two-timescale algorithm: a) *distributive implementation*, and b) *hybrid implementation*.

Under distributive implementation, at each frame, each BS and RS node $m \in \mathcal{R} \cup \{0\}$ acquires the local CSI $\{h_j\}_{j \in \mathcal{L}^+(m)}$ and exchange the local control variables $\{p_j\}_{j \in \mathcal{L}^+(m)}$, $\{\lambda_i^{(m)}\}$ and $\{r_j\}_{j \in \mathcal{L}^+(m)}$ with neighbor nodes. It then updates the long-term flow control $\{r_j\}_{j \in \mathcal{L}^+(m)}$ once according to the outer iteration (55), and updates the power control variables $\{p_j\}_{j \in \mathcal{L}^+(m)}$ and $\{\lambda_i^{(m)}\}$ in each time slot according to the inner iterations (53) and (54). As an illustrative example, Fig. 6. a) demonstrates the message passing under distributive implementation and the network topology in Fig. 1.

Under hybrid implementation, there is a RRM server coordinating the message passing and the outer loop iterations in the network as illustrated in Fig. 1.. At the beginning of each frame, each BS and RS node $m \in \mathcal{R} \cup \{0\}$ obtains long-term flow control $\{r_j\}_{j \in \mathcal{L}^+(m)}$ from the RRM server and acquires the local CSI $\{h_j\}_{j \in \mathcal{L}^+(m)}$. It then updates the local power control variables $\{p_j\}_{j \in \mathcal{L}^+(m)}$ and $\{\lambda_i^{(m)}\}$ according to the inner iterations (53) and (54) in each time slot within the frame. At the end of the frame, it passes the local variables $\{p_j\}$ and $\{\lambda_i^{(m)}\}$ together with the local CSI $\{h_j\}$ for $j \in \mathcal{L}^+(m)$ to the RRM server. By collecting the short term variables \mathbf{p} and λ as well as the global CSI \mathbf{h} , the RRM server updates the long-term flow control \mathbf{r} using the outer iteration (55) and feeds back to the BS and RSs at the beginning of the next frame. Fig. 6.b) illustrates the message passing under hybrid implementation and the network topology in Fig. 1.

Note that as the inner iterations (53) and (54) require only local CSI, they can be iterated for a finite number of steps $N_s > 1$ at each frame to catch up with the fast timescale CSI variations. On

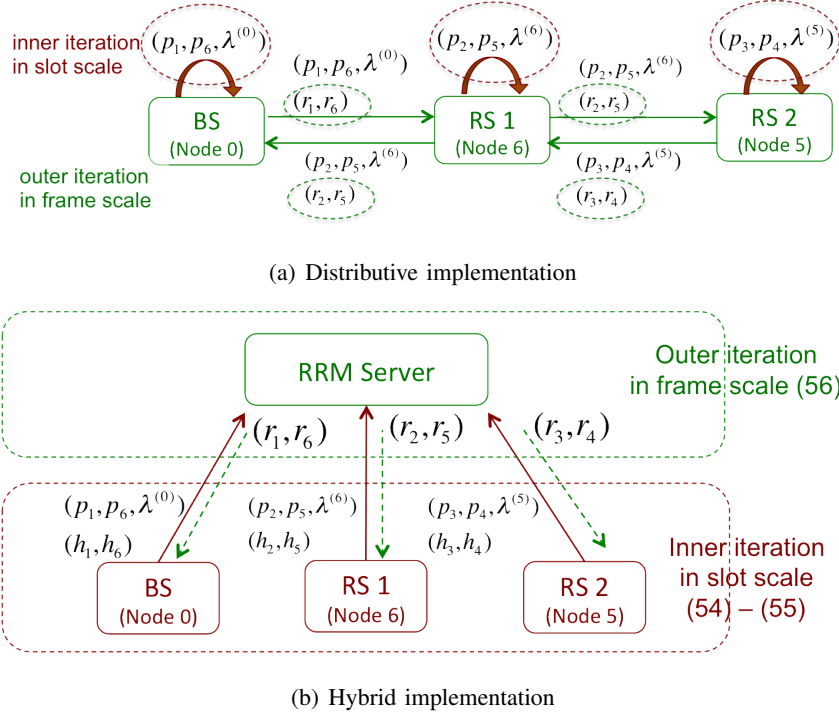


Figure 6. Example algorithm implementations and message passing for the network topology in Fig. 1.

the other hand, since the outer iteration (55) requires global coordination which involves signaling latency, it can only be updated once at each frame. However, as the long term flow control \mathbf{r} adapts to CSI statistics (i.e., the long term CSI \mathbf{h}^l), it does not require a fast iteration and is not sensitive to signaling latency.

Considering the computational complexity, the two-timescale algorithm with variable partitions reduces the computational cost at the central controller by distributing the computation of the cross-layer network utility optimization to different nodes locally. Table III gives a comparison on the computational complexity in terms of CPU time over one frame under the example in Section II-B. The inner iteration is assumed to update for $N_s = 30$ steps in each frame under two-timescale algorithms. The one-timescale centralized algorithm consumes more CPU time, which may not be a good choice, since the transmission power control is delay-sensitive.

C. The Convergence Analysis

In this subsection, we derive the upper bound of the tracking error of the two-timescale iterations (53)-(55) using the theoretical results developed in Section III. We derive the convergence rate parameters α_x and α_y as follows.

Theorem 7 (Local convergence speed): Denote

$$M_L(\cdot) = \begin{pmatrix} \frac{\partial^2}{\partial \mathbf{p} \partial \mathbf{p}} L(\cdot) & \frac{\partial^2}{\partial \mathbf{p} \partial \lambda} L(\cdot) \\ -\frac{\partial^2}{\partial \lambda \partial \mathbf{p}} L(\cdot) & \mathbf{0}_{J \times J} \end{pmatrix}.$$

	RRM Server (ms)	Each BS (RS) (ms)
Two-timescale distributive algorithm	-	0.601
Two-timescale hybrid algorithm	0.0511	0.533
One-timescale centralized algorithm	1.26	-

Table III

COMPUTATIONAL COMPLEXITY IN TERMS OF CPU TIME OF THE ONE-TIMESCALE CENTRALIZED ALGORITHM AND TWO-TIMESCALE ALGORITHMS OVER ONE FRAME. THE SIMULATION WAS DONE ON A MATLAB PLATFORM RUNNING ON A DESKTOP COMPUTER WITH A 2.8 GHZ SINGLE CORE CPU.

Then $M_L(\cdot)$ is negative definite for all $(\mathbf{p}^*, \mathbf{r}^*)$ and $\lambda^*(\mathbf{p}^*)$, under all \mathbf{h} . In addition, given any optimal points $\mathbf{w} = (\mathbf{p}^*, \mathbf{r}^*)$, we have the local convergence rate $\alpha_x(\mathbf{w}) \geq -\lambda_{\max}(\frac{1}{2}(M_L(\mathbf{w}) + M_L^T(\mathbf{w})))$, $\alpha_y(\mathbf{w}) = -\lambda_{\max}\left(\frac{\partial^2}{\partial \mathbf{r} \partial \mathbf{r}} L(\mathbf{w})\right)$, where $\lambda_{\max}(A)$ denotes the maximum eigenvalue of matrix A . ■

Proof: Please refer to Appendix H for the proof. ■

Using Theorem 7, a lower bound of the global convergence rate can be obtained by⁶ $\alpha_x = \inf\{\alpha_x(\omega) : \omega \in \mathcal{X}(y) \times \mathcal{Y}, \forall y \in \mathcal{Y}, \forall \mathbf{h}\}$ and $\alpha_y = \inf\{\alpha_y(\omega) : \omega \in \mathcal{X}(y) \times \mathcal{Y}, \forall y \in \mathcal{Y}, \forall \mathbf{h}\}$.

Given the results in Theorem 7, the condition for algorithm stability and tracking error bound then directly follows from the results in Theorem 4 and 5, respectively.

Note that we can always enhance the convergence and increase α_x and α_y by introducing a carefully chosen positive definite scaling matrix Γ in the iterations. However, the computation of the scaling matrix may increase the complexity for the inner iteration and require additional signaling overhead for the outer iteration.

VII. NUMERICAL RESULTS

In this section, we simulate the tracking performance of the mixed timescale algorithm for the example cross-layer stochastic optimization problem studied in Section II-B and Section VI. We demonstrate the performance advantage of the mixed timescale algorithm over one-timescale algorithms under the CSI model discussed in Section II-A2. In addition, we show that the proposed two-timescale compensation algorithm in Section V significantly reduces the tracking error under time-varying CSI.

We consider the wireless heterogeneous relay network described in Section II-B. Specifically, the network has 1 macro BS, 2 RSs and 4 mobile users who want to transmit data flows to the macro BS.

⁶In fact, the domain $\mathcal{X}(y) \times \mathcal{Y}$ may not be compact, and α_x (or α_y) may then be degenerated. However, in practice, the control variables x and y (corresponding to power and flow data rate here) cannot go unbounded. Therefore, one can identify a confident domain $\overline{\mathcal{X}}(y) \times \overline{\mathcal{Y}} \subseteq \mathcal{X}(y) \times \mathcal{Y}$ and $\overline{\mathcal{H}} \subseteq \mathcal{H}$, which are compact, to estimate the lower bound of the convergence rate α_x (or α_y).

The BSs are static and the mobiles are moving around with a speed at most $v_{\max} = 100$ km/h. The mobility is according to the Levy walk mobility model in Section II-A1 with parameter $D_{\min} = 75$ m, $\iota = 1.8$ and $c_0 = D_{\min}^\iota$. There are 6 wireless links as illustrated in Fig. 1, and it is assumed that the network topology does not change during the simulation. Correspondingly, the long-term CSI timescale parameter is $\epsilon = 6 \times 10^{-4} \text{ sec}^{-1}$. The control objective is to determine the flow rate congestion control \mathbf{r} and power allocation \mathbf{p} according to the proportional fair utility in (13). The frame duration is $\tau = 1$ ms, and the inner iterations (53) and (54) are updated for $N_s = 30$ steps in each frame.

We consider the following baseline schemes:

- **Baseline 1 - One-timescale centralized algorithm based on real-time global CSI** [1], [12]: The central controller (RRM server in Fig. 1) solves the deterministic version (dropping the expectation) of the problem (13) at each time slot. The controller collects real-time global CSI (GCSI) $\mathbf{h}(t)$ at each time slot and computes the optimal flow rate congestion control $\mathbf{r}(\mathbf{h}(t))$ and power allocation $\mathbf{p}(\mathbf{h}(t))$ that adapt to each realization of $\mathbf{h}(t)$.
- **Baseline 2 - One-timescale centralized algorithm based on statistical CSI** [15]: For every $T_s = 100$ ms, the central controller solves a relaxed version of the stochastic optimization problem (13), where the link capacity constraint (14) is replaced by the probability outage constraint $\Pr[(14) \text{ is not satisfied}] \leq \Theta_{\text{out}}$, and the flow rate congestion control \mathbf{r} and power allocation \mathbf{p} adapt to the statistics of the GCSI $\mathbf{h}(t)$.
- **Baseline 3 - Two-timescale stochastic gradient without compensations**: The algorithm iterations are based on stochastic gradient in (23) and (24) in solving Problem 4.

Note that the baseline 1 suffers from huge computational complexity, as it searches for the optimal solution at each time slot, which is not scalable to large networks. Moreover, baseline 1 is very sensitive to signaling latency for the message passing throughout the network⁷. On the other hand, baseline 2 is not sensitive to the signaling latency but it is too conservative as it does not exploit the local real-time CSI knowledge at the BS and the RSs. Hence, baseline 1 and baseline 2 are for performance benchmark only.

A. Performance of the Mixed Timescale Algorithms

Due to the exogenous stochastic variation of $h^s(t)$ and $h^l(t)$, the instantaneous link capacity constraint in (14) and (15) may not be satisfied for every iteration outputs. To quantify the associated

⁷In the current practical communication networks, such as LTE, the backhaul latency is typically around 10-20 ms [42].

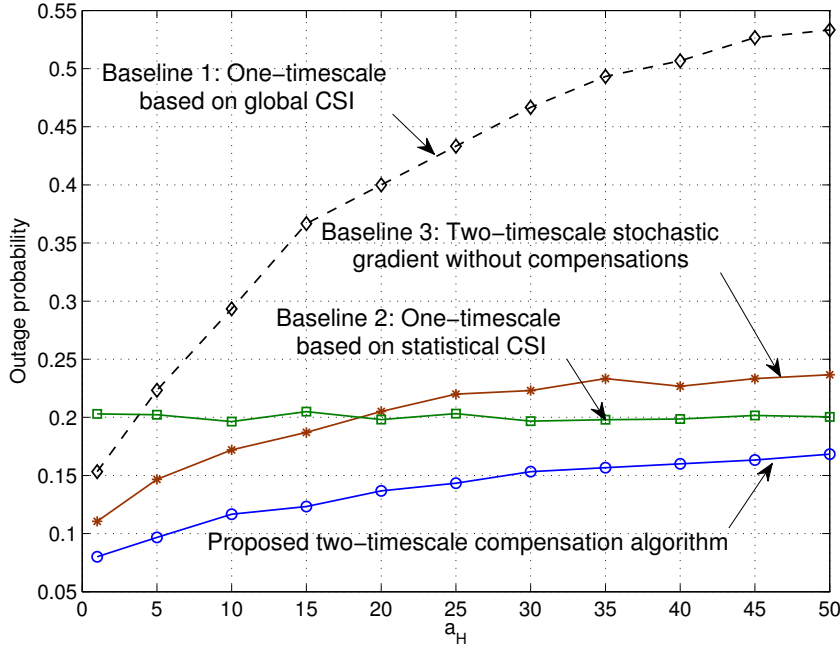


Figure 7. The constraint outage probability under different CSI fading parameters a_H and $\epsilon = 6 \times 10^{-4}$. The signaling latency is $\tau = 5$ ms.

performance penalty, we define the constraint outage probability as follows

$$\mathbb{P}_{\text{out}} = \frac{1}{N_T} \sum_{n=1}^{N_T} \sum_{j \in \mathcal{L}} 1 \{ r_j \notin \mathcal{C}_m^{\text{cap}}(\mathbf{p}(n), \mathbf{h}(n)), \forall m : j \in \mathcal{L}^+(m) \}$$

where N_T is the total number of transmission frames, $1\{\cdot\}$ is the indicator function, and $\mathcal{C}_m^{\text{cap}}(\mathbf{p}, \mathbf{h})$ is the multi-access channel (MAC) capacity region at receiver node m , and is specified by (12).

Note that, $a_H = 50$ corresponds to around 10 ms channel coherence time [31] and $a_H = 1$ yields over 200 ms channel coherence time. Fig. 7 shows the constraint outage probability under different CSI fading parameters a_H and $\epsilon = 6 \times 10^{-4}$. The constraint outage probability increases when the channel is changing faster, but the proposed two-timescale compensation algorithm has the least constraint outage probability compared with other baselines under 5 ms signaling latency and various channel fading rates.

Fig. 8.a) gives the throughput performance assuming no signaling latency. Baseline 1 yields the best performance, but it is highly sensitive to signaling latency, as shown in Fig. 8.b), where 5 ms signaling latency is considered. In Fig. 8.b), as the CSI varies faster, the throughput performance of all the schemes decrease, except for baseline 2. However, baseline 2 does not exploit the short-term transmission opportunity and achieves only moderate performance. As a comparison, the proposed two timescale compensation algorithm has the best performance and is robust to signaling latency. Fig.

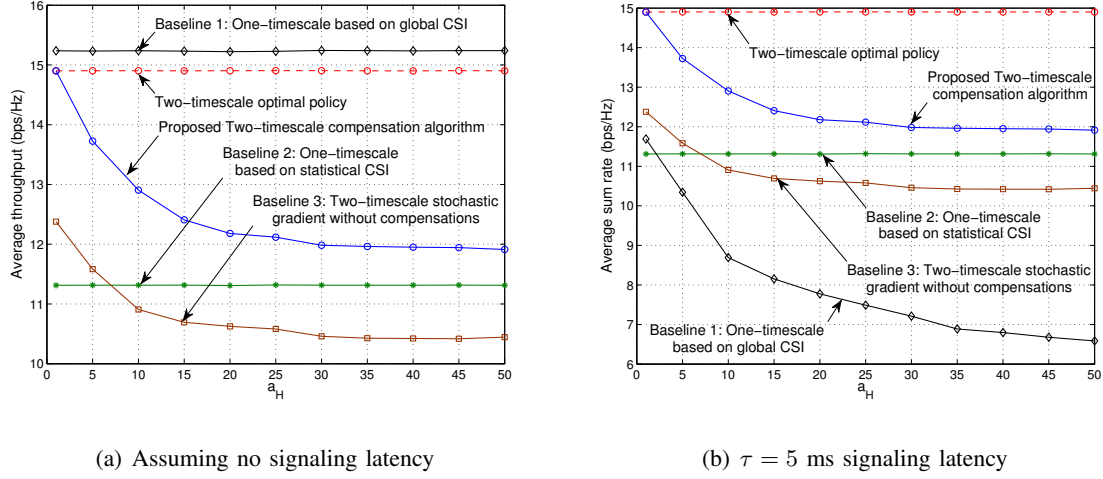


Figure 8. Throughput performance of the different schemes under $\epsilon = 6 \times 10^{-4}$ and average SNR 11 dB. Note that, $a_H = 50$ corresponds to around 10 ms channel coherence time [31] and $a_H = 1$ yields over 200 ms channel coherence time.

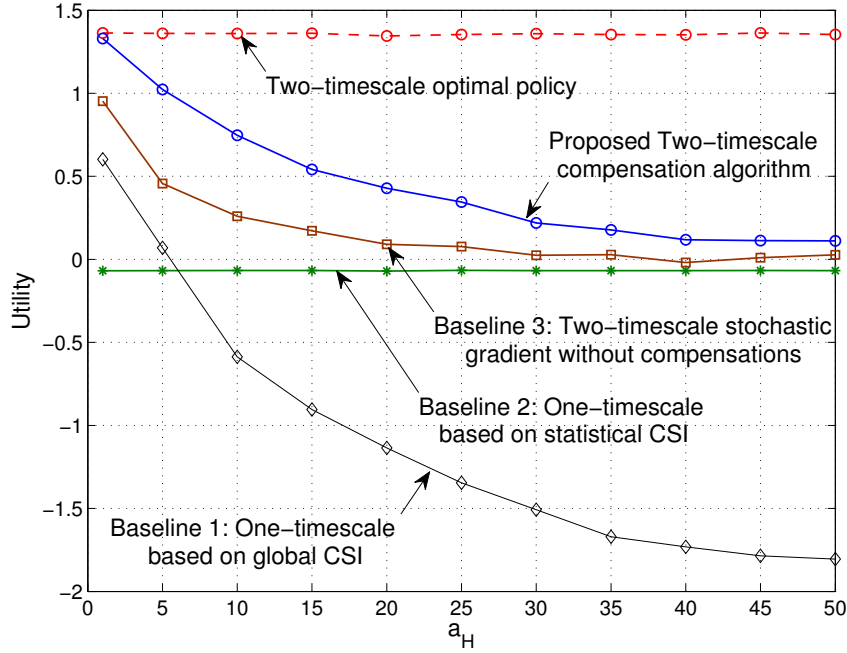


Figure 9. Proportional fair utility of the different schemes under 11 dB average SNR, 5 ms signaling latency and the long timescale CSI parameter $\epsilon = 6 \times 10^{-4}$. The proposed algorithm performs much better than all the baseline schemes.

9 demonstrates the corresponding proportional fair utility for the different schemes under signaling latency of 5 ms. The proposed algorithm performs much better than all the other schemes.

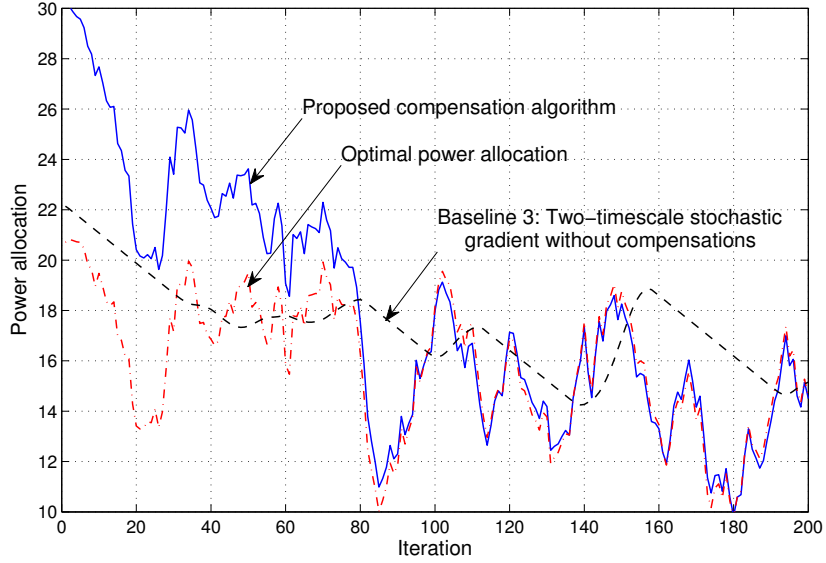


Figure 10. A snapshot of algorithm trajectories of the proposed compensation algorithm and the stochastic gradient algorithm without compensations under short timescale CSI fading parameter $a_H = 10$ and long timescale parameter $\epsilon = 6 \times 10^{-4}$. The trajectories represent the online power allocation policy p_5 . The proposed compensation algorithm quickly converges to the optimal trajectory of the inner iteration, while the baseline algorithm fails to track the optimal target and yields much larger tracking errors.

B. Tracking Performance of the Adaptive Compensation Algorithm

We evaluate the tracking performance of the two-timescale compensation algorithm over the baseline stochastic gradient algorithm.

Fig. 10 shows a snapshot of the algorithm trajectories of the proposed two-timescale compensation algorithm and the baseline stochastic gradient tracking algorithm without compensations, under short timescale CSI fading rate $a_H = 10$ and long timescale parameter $\epsilon = 6 \times 10^{-4}$. The trajectories represent the online power allocation policy p_5 . The proposed compensation algorithm quickly converges to the optimal trajectory of the inner iteration, while the baseline algorithm fails to track the optimal target and yields much larger tracking errors.

VIII. CONCLUSIONS

In this paper, we have analyzed the convergence behavior of a mixed timescale cross-layer stochastic optimization driven by multi-timescale CSI. The CSI dynamic is modeled by an auto-regressive process in the short timescale (small-scale fading), and a mobility-driven dynamic process in the long timescale (large-scale fading). We partitioned the control variables into short-term control variables and long-term control variables, and studied the convergence of the corresponding mixed timescale

stochastic iterative algorithm. We derived a VSDS and showed that studying the algorithm convergence is equivalent to studying the stochastic stability of the VSDS. Using Lyapunov stochastic stability analysis, we derived a sufficient condition for the algorithm to be stable. In addition, we derived a tracking error upper bound in terms of the parameters of the mixed timescale CSI process. Based on these results, we proposed an adaptive compensation algorithm for enhancing the tracking performance. The analysis framework and the proposed algorithms were applied to an application example in a wireless heterogeneous network. Numerical results matched with the theoretical insights and demonstrated significant performance gain of the proposed compensation algorithms over the baselines.

APPENDIX A

DERIVATIONS OF THE REFLECTION TERMS z_x AND z_y

Taking a small step Δt , the ODE dynamics (30)-(31) can be written as

$$\begin{aligned} x_c(t + \Delta t) &= x_c(t) + G(x_c(t), y_c(t), h^s(t), h^l(t))\Delta t + z_x(t)\Delta t \\ &= \mathcal{P}_{\mathcal{X}(y)} \left[x_c(t) + G(x_c(t), y_c(t), h^s(t), h^l(t))\Delta t \right] \\ y_c(t + \Delta t) &= y_c(t) + k(y_c(t), h^l(t))\Delta t + z_y(t)\Delta t \\ &= \mathcal{P}_{\mathcal{Y}} \left[y_c(t) + k(y_c(t), h^l(t))\Delta t \right]. \end{aligned}$$

Consider that the convex domains $\mathcal{X}(y)$ and \mathcal{Y} can be specified by a set of constraints $\omega_i(x, y; h) \leq 0$, $i = 1, \dots, W$, and $q_i(y; h^l) \leq 0$, $i = 1, \dots, J$, respectively. Then the Euclidean projections are equivalent to find the points $x_0(\Delta t)$ and $y_0(\Delta t)$, which solve the following minimization problems

$$\begin{aligned} \min_x \quad & \frac{1}{2} \|x - (x_c(t) + G(x_c(t), y_c(t), \cdot)\Delta t)\|_2^2 \\ \text{subject to} \quad & \omega_i(x, y; \cdot) \leq 0, \quad \forall i = 1, \dots, W. \end{aligned} \tag{56}$$

and

$$\begin{aligned} \min_y \quad & \frac{1}{2} \|y - (y_c(t) + k(y_c(t), \cdot)\Delta t)\|_2^2 \\ \text{subject to} \quad & q_i(y; \cdot) \leq 0, \quad \forall i = 1, \dots, J. \end{aligned} \tag{57}$$

The corresponding Lagrange functions are given by

$$L^{(x)}(x, y, \lambda_x) = \frac{1}{2} \|x - [x_c(t) + G(\cdot)\Delta t]\|_2^2 + \sum_{i=1}^W \lambda_{x,i} \omega_i(x, y; \cdot)$$

and

$$L^{(y)}(y, \lambda_y) = \frac{1}{2} \|y - [y_c(t) + k(\cdot)\Delta t]\|_2^2 + \sum_{i=1}^J \lambda_{y,i} q_i(y; \cdot).$$

The KKT condition [32], [33] for the problem (56) on the x variable is given by

$$x_0(\Delta t) - [x_c(t) + G(x_c(t), y_c(t), \bullet)\Delta t] + \sum_{i=1}^W \lambda_{x,i}^* \nabla_x \omega_i(x_0(\Delta t), y_c(t); \bullet) = 0 \quad (58)$$

$$\lambda_{x,i}^* \omega_i(x_0(\Delta t), y_c(t); \bullet) = 0, \forall i. \quad (59)$$

Solving (58)-(59), we obtain $x_0(\Delta t)$. Similarly, by writing the KKT condition for (57), we can obtain $y_0(\Delta t)$. Then the reflection terms are given by

$$z_x(t) = \lim_{\Delta t \rightarrow 0} \frac{x_0(\Delta t) - [x_c(t) + G(x_c(t), y_c(t), \bullet)\Delta t]}{\Delta t}$$

and

$$z_y(t) = \lim_{\Delta t \rightarrow 0} \frac{y_0(\Delta t) - [y_c(t) + k(y_c(t), \bullet)\Delta t]}{\Delta t}.$$

APPENDIX B

SKETCH PROOF OF THEOREM 2

We ignore the transient states for x_{n_s} and y_{n_f} , and just focus on their stationary states.

For the inner iteration x_{n_s} in (21), consider a large enough n_s . Since μ_{n_f} is decreasing, we have $\mu_{n_f} \ll \gamma$. From the timescale condition $a_H \tau \ll \gamma$ and the step size condition $\mu_{n_f} \ll \gamma$, the iteration (21) finds the partial optimum $\hat{x}(y, h(n_f \tau))$ for each $h(t) = (h_1(t), \dots, h_N(t))$, where $h_j(t) = h_j^l h_j^s(t)$. This can be shown under Assumption 3 and 4 given a sufficiently small step size γ [32], [33]. Note that the partial stationary point $\hat{x}(y, h(n_f \tau))$ of (21) is also the partial equilibrium point $x_c(t)$ of (30). Therefore, we have established that $x_{n_s(t)} \rightarrow x_c(t)$, where $n_s(t) = \lfloor tN_s/\tau \rfloor$.

For the outer iteration y_{n_f} in (22), with sufficient large n_f , we have

$$\begin{aligned} \mathbb{E} [K(x_{n_s-1}, y_{n_f-1}; h^s(n_f \tau), h^l | h^l)] &= \mathbb{E} [\Gamma_y \nabla_y F(\hat{x}(y_{n_f-1}, \bullet), y_{n_f-1}; h^s(n_f \tau), h^l) | h^l] \\ &= \mathbb{E} [\Gamma_y \nabla_y \mathcal{P}_1(y, h^s, h^l) | h^l] \\ &= \Gamma_y \int \nabla_y \mathcal{P}_1(y, \omega, h^l) dF_{h^s}(\omega) \\ &= \Gamma_y \nabla_y \int \mathcal{P}_1(y, \omega, h^l) dF_{h^s}(\omega) \\ &= \Gamma_y \nabla_y \mathbb{E} [\mathcal{P}_1(y, h^s, h^l) | h^l] \\ &\triangleq \bar{K}(y, h^l) \end{aligned} \quad (60)$$

where $F_{h^s}(\omega)$ is the cumulative distribution function of the conditional probability $\Pr(h^s = \omega | h^l) = \Pr(h^s = \omega)$, and the interchange of the integration and the differentiation is because the integration is bounded (i.e., the expectation of $F(\cdot)$ is bounded). We take conditional expectation here because y adapts to each realization of h^l . Therefore, we have the gradient estimator $K(\cdot) = \bar{K}(y, h^l) + \xi(\cdot)$, where $\xi(\cdot)$ is some “noise” and $\mathbb{E}\xi(\cdot) = 0$. Using the stochastic approximation [19], [36], y_n converges to y^* almost surely under the assumed step size rule for μ_n .

From Assumption 4, the matrix $\nabla_y k(y, h^l) = \nabla_y \mathbb{E}[K(\cdot)] = \mathbb{E}[\nabla_y K(\cdot)]$ is a negative definite matrix. Therefore, the system $\dot{y}_c = \nabla_y k(y_c, h^l)$ is asymptotically stable [39] at a unique stationary point $y_c^*(h^l)$, where $\dot{y}_c^*(h^l) = \nabla_y k(y_c, h^l) = 0$. According to the definition of $k(y, h^l)$, we have $\bar{K}(y, h^l) \equiv k(y, h^l)$. This implies that the stationary point y_n of (22) is just the equilibrium y_c^* of (31), i.e., $y^* = y_c^*$. Then we have established the asymptotic result for $\limsup_{t \rightarrow \infty} \Pr \{ \|y_{n_f(t)} - y_c(t)\| > \eta \} = 0$, where $n_f(t) = \lfloor t/\tau \rfloor$.

APPENDIX C

PROOF OF LEMMA 1

The above results are obtained from the implicit function theorem. From the optimality condition $\tilde{G}(\hat{x}_c, y_c, h^s, h^l) = 0$ and the implicit function theorem, we have

$$\begin{aligned} d\hat{x}_c(y_c, \cdot) &= \tilde{G}_x^{-1}(\hat{x}_c(y_c, \cdot), y_c, h^s, h^l) \left[\tilde{G}_{h^s}(\hat{x}_c(y_c, \cdot), y_c, h^s, h^l) dh^s \right. \\ &\quad \left. + \tilde{G}_{h^l}(\hat{x}_c(y_c, \cdot), y_c, h^s, h^l) dh^l + \tilde{G}_y(\hat{x}_c(y_c, \cdot), y, h^s, h^l) dy \right]. \end{aligned}$$

But since $dh^l \ll dh^s$ due to the small variation of $h^l(t)$ (controlled by $\epsilon \ll a_H$), the term $G_{h^l}(\cdot)dh^l$ is comparatively small. Ignoring this term, equation (33) yields.

APPENDIX D

PROOF OF THEOREM 3

Theorem 3 can be obtained from the weak convergence results in [19], [36], [43]. In the following, we sketch briefly how we can apply those results.

Recall that the algorithm is implemented on the timescale $t_{n_f} = n_f \tau$, where n_f is the frame index and τ is the frame duration. To establish the VSDS, we define the virtual algorithm timescales as follows.

Definition 11 (Virtual algorithm timescale): The virtual algorithm timescale on frame is a mapping from the frame index n_f to a real number $s_{n_f} = n_f N_s \gamma$. The virtual algorithm timescale on slot is a mapping from the slot index n_s to a real number $\tilde{s}_{n_s} = n_s \gamma$. ■

Under the virtual algorithm timescale, the iteration indices are related as $n_s(s) = \lfloor s/\gamma \rfloor$ and $n_f(s) = \lfloor s/(N_s \gamma) \rfloor$. The virtual algorithm times is just a scaled implementation time, and their relationship is given by $t_{n_f} = N_s n_f \gamma \frac{\tau}{N_s \gamma} = s_{n_f} \frac{\tau}{N_s \gamma}$.

Accordingly, denote the virtual CSI state as $\tilde{h}^s(s)$. Since $dt_{n_f} = \frac{\tau}{N_s \gamma} ds_{n_f}$, the timescale of the virtual CSI dynamics $\tilde{h}^s(s)$ can be aligned with the virtual algorithm timescale s_{n_f} by adding a gain parameter $\frac{\tau}{N_s \gamma}$ to (1) as,

$$d\tilde{h}^s = -\frac{1}{2} \frac{a_H \tau}{N_s \gamma} \tilde{h}^s ds + \sqrt{\frac{a_H \tau}{N_s \gamma}} dW_s. \quad (61)$$

$\tilde{h}^s(s)$ has the same shape as $h^s(t)$, but with a different timescale.

The same trick applies to the long-term virtual CSI dynamics $\tilde{h}^l(s)$, as $d\tilde{h}_j^l = -\frac{\tau}{N_s\gamma}c_0\iota D_j(s)^{-\iota-1}v_j(s)ds$, $\forall j$. In a vector form, we have

$$d\tilde{h}^l = -\frac{\tau}{N_s\gamma}H_L(t)ds \quad (62)$$

where $H_L(t)$ is an $N \times N$ diagonal matrix, with the j -th diagonal element being $c_0\iota D_j(t)^{-\iota-1}v_j(t)$.

The virtual CSI timescale separation parameter becomes $\tilde{\epsilon} = \frac{\tau}{N_s\gamma}\epsilon$.

We use a localization method [43] and consider the algorithm trajectories (21)-(22) and (30)-(31) start from (x_0, y_0) at time $s = s_0 = 0$, which lies in the neighborhood of $(x^*(\tilde{h}^s, \tilde{h}^l), y^*(\tilde{h}^l))$. We have

$$\begin{aligned} & x_{n_s} - x_c(\tilde{s}_{n_s}) \\ = & x_{n_s-1} - x_c(\tilde{s}_{n_s-1}) \\ & + \gamma \left[G(x_{n_s-1}, y_{n_f}, \tilde{h}^s(s_{n_f}), \tilde{h}^l(s_{n_f})) + z_{x,n_s} - G(x_c(\tilde{s}_{n_s-1}), y_c(s_{n_f}), \tilde{h}^s(s_{n_f}), \tilde{h}^l(s_{n_f})) - z_x(\tilde{s}_{n_s-1}) \right] \\ & - \left[x_c(\tilde{s}_{n_s}) - x_c(\tilde{s}_{n_s-1}) - \gamma G(x_c(\tilde{s}_{n_s-1}), y_c(s_{n_f}), \tilde{h}^s(s_{n_f}), \tilde{h}^l(s_{n_f})) - z_x(\tilde{s}_{n_s}) \right] \end{aligned} \quad (64)$$

where using Taylor expansion,

$$\begin{aligned} \text{term (63)} = & \gamma G_x \left(x_c(\tilde{s}_{n_s-1}), y_c(s_{n_f}), \tilde{h}^s(s_{n_f}), \tilde{h}^l(s_{n_f}) \right) (x_{n_s-1} - x_c(\tilde{s}_{n_s-1})) \\ & - \gamma G_y \left(x_c(\tilde{s}_{n_s-1}), y_c(s_{n_f}), \tilde{h}^s(s_{n_f}), \tilde{h}^l(s_{n_f}) \right) (y_{n_f} - y_c(s_{n_f})) + o(\gamma). \end{aligned}$$

Here, it is reasonable to consider $z_{x,n_s} - z_x(\tilde{s}_{n_s}) = 0$, since if the partial equilibrium $\hat{x}_c \in \mathcal{X}$, we eventually have $z_{x,n_s} = z_x(\tilde{s}_{n_s}) = 0$. If $\hat{x}_c \in \partial\mathcal{X}$, both trajectories eventually search along the boundary, and $z_{x,n_s} = z_x(\tilde{s}_{n_s})$ for a large enough n_s .

The term (64) is just a first order Taylor expansion of the continuous trajectory (30) at the point $x_c(\tilde{s}_{n_s-1})$. By taking $\tilde{x}_c^\gamma(\tilde{s}_{n_s}) = \frac{1}{\sqrt{\gamma}}(x_{n_s} - x_c(\tilde{s}_{n_s}))$ and $\tilde{y}_c^\gamma(s_{n_f}) = \frac{1}{\sqrt{\gamma}}(y_{n_f} - y_c(s_{n_f}))$, we have

$$\begin{aligned} \tilde{x}_c^\gamma(\tilde{s}_n) &= \tilde{x}_c^\gamma(0) + \gamma \sum_{j=0}^{n-1} \left[G_x \left(x_c(\tilde{s}_j), y_c(s_j), \tilde{h}^s(s_j), \tilde{h}^l(s_j) \right) \tilde{x}_c^\gamma(\tilde{s}_j) \right. \\ &\quad \left. + G_y \left(x_c(\tilde{s}_j), y_c(s_j), \tilde{h}^s(s_j), \tilde{h}^l(s_j) \right) \tilde{y}_c^\gamma(s_j) \right] + o(\gamma) \\ &= \tilde{x}_c^\gamma(0) + \int_0^{\tilde{s}_n} (G_x(\bullet)\tilde{x}_c^\gamma + G_y(\bullet)\tilde{y}_c^\gamma) ds + o(\gamma) \end{aligned}$$

where $\underline{j} \triangleq \lfloor j/N_s \rfloor$ frame index of the outer iteration when the inner iteration is at the j -th slot. Equivalently,

$$d\tilde{x}_c^\gamma = G_x(\bullet)\tilde{x}_c^\gamma ds + G_y(\bullet)\tilde{y}_c^\gamma ds + o(\gamma). \quad (65)$$

Similarly, we derive the dynamic $\tilde{y}_c^\gamma(s)$ as follows.

$$\begin{aligned}
& y_{n_f} - y_c(s_{n_f}) \\
= & y_{n_f-1} - y_c(s_{n_f-1}) \\
& + N_s \gamma \left[N_s^{-1} K(x_{\overline{n_f}}, y_{n_f-1}; \bullet) + z_{y, n_f} - N_s^{-1} K(\hat{x}_c(s_{n_f-1}), y_c(s_{n_f-1}); \bullet) - \hat{z}_y(s_{n_f}) \right] \quad (66) \\
& + \gamma \left[K(\hat{x}_c(s_{n_f-1}), y_c(s_{n_f-1}); \bullet) + \hat{z}_y(s_{n_f}) - k(y_c(s_{n_f-1}), \tilde{h}^l(s_{n_f})) - z_y(s_{n_f}) \right] \quad (67) \\
& - \left[y_c(s_{n_f}) - y_c(s_{n_f-1}) - \gamma k(y_c(s_{n_f-1}), \tilde{h}^l(s_{n_f})) - \gamma z_y(s_{n_f}) \right] \quad (68)
\end{aligned}$$

where $\overline{n_f} \triangleq N_s n_f$ is the slot index of the inner iteration when the outer iteration is at the n_f -th frame. The term (66)

$$\begin{aligned}
= & (N_s \gamma) N_s^{-1} \left[K(x_{\overline{n_f}}, y_{n_f-1}; \bullet) - K(\hat{x}_c(s_{n_f-1}), y_{n_f-1}; \bullet) \right. \\
& \left. + K(\hat{x}_c(s_{n_f-1}), y_{n_f-1}; \bullet) - K(\hat{x}_c(s_{n_f-1}), y_c(s_{n_f-1}); \bullet) \right] \\
= & (N_s \gamma) N_s^{-1} K_x(\hat{x}_c(s_{n_f-1}), y_c(s_{n_f-1}); \bullet) (x_{\overline{n_f}-1} - x_c(s_{n_f-1}) + x_c(s_{n_f-1}) - \hat{x}_c(s_{n_f-1})) \\
& + (N_s \gamma) N_s^{-1} K_y(\hat{x}_c(s_{n_f-1}), y_c(s_{n_f-1}); \bullet) (y_{n_f-1} - y_c(s_{n_f-1})) + o(\gamma) \\
= & (N_s \gamma) N_s^{-1} K_x(\hat{x}_c(s_{n_f-1}), y_c(s_{n_f-1}); \bullet) (\tilde{x}_c^\gamma + \tilde{x}_c^{e, \gamma}) \\
& + (N_s \gamma) N_s^{-1} K_y(\hat{x}_c(s_{n_f-1}), y_c(s_{n_f-1}); \bullet) \tilde{y}_c^\gamma + o(\gamma)
\end{aligned}$$

by the first order Taylor expansion of function $K(\cdot)$ at $(\hat{x}_c(s_{n_f-1}), y_c(s_{n_f-1}))$. Note that, since $\Delta s = s_{n_f} - s_{n_f-1} = N_s \gamma$,

$$\frac{1}{\sqrt{\gamma}} \text{term (66)} = N_s^{-1} K_x(\hat{x}_c, y_c, \bullet) (\tilde{x}_c^\gamma + \tilde{x}_c^{e, \gamma}) \Delta s + N_s^{-1} K_y(\hat{x}_c, y_c, \bullet) \tilde{y}_c^\gamma \Delta s + o(\gamma). \quad (69)$$

Consider $y^*(\tilde{h}^l) \in \mathring{\mathcal{Y}}$ is in the interior of the domain. (The case when $y^*(\tilde{h}^l)$ is on the boundary will be discussed later). Then it is reasonable to consider $y_{n_f} \in \mathring{\mathcal{Y}}$, with probability 1. We consider the following process for the term (67),

$$S^\gamma(s) := \gamma \sum_{j=1}^{n_f(s)} \left[K(\hat{x}_c(s_j), y_c(s_j); \bullet) - k(y_c(s_j), \tilde{h}^l) \right].$$

Choosing a sufficiently small $\delta > 0$, we have

$$S^\gamma(s + \delta) - S^\gamma(s) \approx \gamma \sum_{j=n_f(s)+1}^{n_f(s+\delta)} \left[K(\hat{x}_c(s_j), y; \bullet) - k(y; \tilde{h}^l) \right].$$

where $y = y_c(s)$. The above is an approximation since we use $y_c(s_j) \approx y_c(s)$ for $n_f(s) < j \leq n_f(s + \delta)$. However, the approximation is asymptotically accurate for sufficiently small δ and γ . The central limit theorem for the state dependent process suggests that $\hat{\Sigma}_{s, \delta}^{-1/2} (S^\gamma(s + \delta) - S^\gamma(s))$ weakly

converge to a normal random variable, where $\hat{\Sigma}_{s,\delta}$ is the covariance matrix of $S^\gamma(s+\delta) - S^\gamma(s)$ [36]. This implies that

$$\frac{1}{\sqrt{\gamma}} (S^\gamma(s+\delta) - S^\gamma(s)) \rightarrow \int_s^{s+\delta} \tilde{\Sigma}_{s,\delta}^{\frac{1}{2}} dW_u \quad (70)$$

where W_u is a standard Winner process and

$$\tilde{\Sigma}(y_c(s_0); \tilde{h}^l) = \gamma \sum_{j=-\infty}^{+\infty} \text{cov} \left[K \left(\hat{x}_c(s_j), y_c(s_j); \tilde{h}(s_j) \right), K \left(\hat{x}_c(s_0), y_c(s_0); \tilde{h}(s_0) \right) \middle| \tilde{h}^l \right]$$

is the covariance matrix of the estimator $K(\cdot)$ under the virtual long-term CSI state \tilde{h}^l . In addition, we have

$$\begin{aligned} \hat{\Sigma}(y_c; \tilde{h}^l) &\triangleq \sum_{j=-\infty}^{+\infty} \text{cov} \left[K \left(\hat{x}_c(t_j), y_c(t_j); h(t_j) \right), K \left(\hat{x}_c(t_0), y_c(t_0); h(t_0) \right) \middle| \tilde{h}^l \right] \\ &= \frac{N_s}{\tau} \tilde{\Sigma}(y_c; \tilde{h}^l) \end{aligned}$$

to be the covariance matrix of the estimator $K(\cdot)$ under the CSI state h , since the virtual algorithm timescale s_n is $\frac{\tau}{N_s \gamma}$ times denser than the implementation timescale t_n . Therefore, from (69) and (70), we obtain

$$d\tilde{y}_c^\gamma = N_s^{-1} K_x(\hat{x}_c, y_c, \cdot) (\tilde{x}_c^\gamma + \tilde{x}_c^{e,\gamma}) ds + N_s^{-1} K_y(\hat{x}_c, y_c, \cdot) \tilde{y}_c^\gamma ds + \frac{\tau}{N_s} \hat{\Sigma}^{\frac{1}{2}}(y_c; \tilde{h}^l) dW_s + o(\gamma).$$

Note that, in a finite horizon case for $s \in [0, T_s]$, by letting $\gamma \rightarrow 0$, one can drop the $o(\gamma)$ term and obtain the convergence results $(\tilde{x}_c^\gamma, \tilde{y}_c^\gamma) \rightarrow (\tilde{x}_c, \tilde{y}_c)$, where $(\tilde{x}_c, \tilde{y}_c)$ is the solution to (71)-(72). In addition, using the sophisticated techniques in [19], [36], one can further prove the convergence in the infinite horizon case for $s \in [0, \infty)$ and obtain the following,

$$d\tilde{x}_c = G_x(x_c, y_c, \tilde{h}^s, \tilde{h}^l) \tilde{x}_c ds + G_y(x_c, y_c, \tilde{h}^s, \tilde{h}^l) \tilde{y}_c ds \quad (71)$$

$$\begin{aligned} d\tilde{y}_c &= N_s^{-1} K_x(\hat{x}_c, y_c, \cdot) (\tilde{x}_c + \tilde{x}_c^e) ds + N_s^{-1} K_y(\hat{x}_c, y_c, \cdot) \tilde{y}_c ds \\ &\quad + \sqrt{\tau N_s^{-1} \Sigma^{\frac{1}{2}}(y_c, \tilde{h}^l)} dW_s + dZ_y. \end{aligned} \quad (72)$$

where $\Sigma^{\frac{1}{2}}(y_c, \tilde{h}^l) = \hat{\Sigma}^{\frac{1}{2}}(y_c, \tilde{h}^l)$ in the case for $y^*(\tilde{h}^l) \in \mathcal{Y}$.

Moreover, changing the MCTS (30)-(31) into to the virtual algorithm time s , we get

$$dx_c = G(x_c, y_c, \tilde{h}^s, \tilde{h}^l) ds + dZ_x \quad (73)$$

$$dy_c = N_s^{-1} k(y_c, \tilde{h}^l) ds + dZ_y \quad (74)$$

where dZ_x and dZ_y are reflection terms. Notice that $d\tilde{x}_c^e = dx_c - d\hat{x}_c$ and $d\tilde{y}_c^e = dy_c - dy^*$. We obtain the following error dynamic system

$$d\tilde{x}_c^e = G(x_c, y_c, \tilde{h}^s, \tilde{h}^l) ds + dZ_x + \tilde{G}_x^{-1} \tilde{G}_{h^s}(\cdot) d\tilde{h}^s + \tilde{G}_x^{-1} \tilde{G}_y(\cdot) dy_c \quad (75)$$

$$d\tilde{y}_c^e = N_s^{-1} k(y_c, \tilde{h}^l) ds + dZ_y - \psi_{h^l}(\tilde{h}^l) d\tilde{h}^l \quad (76)$$

Consider the SDEs (71)-(72) and (75)-(76), and the virtual CSI dynamics (61)-(62). Rearranging the terms and changing the virtual time notation s to t , we obtain the VSDS in (37). This proves the claimed results.

Remark 5 (The case $y^(\tilde{h}^l)$ on the boundary):* When $y^*(\tilde{h}^l)$ is on the boundary, we can follow the argument in [43] to find out the behavior of $\tilde{y}_c(s)$. Note that the corresponding effect is only on the diffusion term $\Sigma^{\frac{1}{2}}(y_c, \tilde{h}^l)dW_s$, where $\Sigma(y_c, \tilde{h}^l) = \Sigma_0(y_c)\hat{\Sigma}(y_c, \tilde{h}^l)$, and $\Sigma_0(y) = \text{diag}\left(\{\sigma_i^0(y)\}_{i=1}^{N_y}\right)$, which is defined in the following. Consider the i -th component of $y^*(\tilde{h}^l)$ is on the boundary. There are two cases. Case i), the i -th component of the drift $k(y^*; h^l)$ is non-zero, which means there must be a reflection force $z_y^{(i)} \neq 0$ on $y_{n_f}^{(i)}$ and $y_c^{(i)}$ to keep them from reaching out of the boundary. Then obviously, upon reaching $y^{(i)}$, $y_{n_f}^{(i)}$ is not likely to be disturbed by the noise from the estimator $K(\cdot)$ [unless the noise is larger than the drift $k^{(i)}(y^*; h^l)$], and the i -th component of the diffusion term $\Sigma^{\frac{1}{2}}(y_c, \tilde{h}^l)dW_s$ should be zero. Hence $\sigma_i^0 = 0$. Case ii), the i -th component of the drift $k(y^*; h^l)$ is zero, which means the reflection force $z_y^{(i)}$ depends on the disturbance noise. According to [43], we can simply consider $\sigma_i^0 = 1$, just as the case when $y^*(\tilde{h}^l)$ is in the interior of \mathcal{Y} . ■

APPENDIX E

PROOF OF THEOREM 4

A. The Lyapunov Drift of the VSDS

It is equivalent to show the VSDS $u(t)$ is stochastically stable. We first give the following lemma.

Lemma 3 (Ito's lemma): Consider a stochastic process $u(t)$ given by the following SDE, $du = f(u)dt + g(u)dW_t$ and a function $V(u) \in \mathbb{R}_+$. The Lyapunov drift operator on $V(\cdot)$ can be written as $\tilde{\mathcal{L}}V = \frac{\partial V}{\partial u}f(u) + \text{tr}\left[g(u)^T \frac{\partial^2 V}{\partial uu^T}g(u)\right]$. ■

We first consider that the optimal solution y^* is in the interior of \mathcal{Y} , which means the reflection term $dZ_y \equiv 0$ and the matrix Σ^0 (defined in Appendix D) is an identity matrix. Then using Lemma 9, the Lyapunov drift of the stochastic process $u(t)$ can be written as

$$\begin{aligned} \tilde{\mathcal{L}}V(u) = & \begin{bmatrix} \tilde{x}_c \\ \tilde{y}_c \end{bmatrix}^T \begin{bmatrix} G_x(x_c, y_c, \bullet) & G_y(x_c, y_c, \bullet) \\ N_s^{-1}K_x(\hat{x}_c, y_c, \bullet) & N_s^{-1}K_y(\hat{x}_c, y_c, \bullet) \end{bmatrix} \begin{bmatrix} \tilde{x}_c \\ \tilde{y}_c \end{bmatrix} \\ & + \tilde{y}_c^T N_s^{-1}K_x(\hat{x}_c, y_c, \bullet)\tilde{x}_c^e + (\tilde{x}_c^e)^T G(x_c, \bullet) + (\tilde{x}_c^e)^T G_x^{-1}G_{h^s}(-\frac{1}{2})\frac{a_H\tau}{N_s\gamma}\tilde{h}^s \\ & + (\tilde{x}_c^e)^T G_x^{-1}G_y N_s^{-1}k(y_c, \tilde{h}^l) + (\tilde{y}_c^e)^T N_s^{-1}k(y_c, \tilde{h}^l) \\ & + \frac{\tau}{N_s\gamma}(\tilde{y}_c^e)^T \psi_{h^l}(\tilde{h}^l)H_L - \frac{1}{2}\frac{a_H\tau}{N_s\gamma}(\tilde{h}^s)^T \tilde{h}^s + \frac{1}{2}\text{tr}(\tau N_s^{-1}\Sigma(y_c)) \\ & + \frac{1}{2}\text{tr}\left[\frac{a_H\tau}{N_s\gamma}(G_x^{-1}G_{h^s})^T(G_x^{-1}G_{h^s}) + \frac{a_H\tau}{N_s\gamma}\mathbf{I}\right] \end{aligned} \quad (77)$$

where \hat{x}_c denotes the partial optimum $\hat{x}_c(y_c, h^s, h^l; t)$.

B. The Upper Bound of the Drift

We bound each term from the above equation in the following.

First of all, from Assumption 4, we can show that

$$\begin{aligned} Q_c &\triangleq \begin{bmatrix} \tilde{x}_c^T & \tilde{y}_c^T \end{bmatrix} \begin{bmatrix} G_x(x_c, y_c, h) & G_y(x_c, y_c, h) \\ N_s^{-1}K_x(x_c, y_c, h) & N_s^{-1}K_y(x_c, y_c, h) \end{bmatrix} \begin{bmatrix} \tilde{x}_c \\ \tilde{y}_c \end{bmatrix} \\ &\leq -N_s^{-1}\alpha(\|\tilde{x}_c\|^2 + \|\tilde{y}_c\|^2). \end{aligned} \quad (78)$$

Secondly, the mapping $G(x_c, \bullet)$ on the x part has the property

$$\begin{aligned} (\tilde{x}_c^e)^T G(x_c, \bullet) &= (\tilde{x}_c^e)^T \int_0^1 G_x(x_c^* + \xi \tilde{x}_c^e, \bullet) \tilde{x}_c^e d\xi \\ &\leq -\alpha_x \|\tilde{x}_c^e\|^2. \end{aligned} \quad (79)$$

Thirdly, the mean mapping $k(y_c, \tilde{h}^l)$ on the y part satisfies

$$\begin{aligned} (\tilde{y}_c^e)^T k(y_c, \tilde{h}^l) &= (\tilde{y}_c^e)^T \lim_{n \rightarrow \infty} \mathbb{E} [K(y_c, \tilde{h}^l)] \\ &= \lim_{n \rightarrow \infty} \mathbb{E} \left[(\tilde{y}_c^e)^T \int_0^1 K_y(y_c^* + \xi \tilde{y}_c^e, \tilde{h}^l) \tilde{y}_c^e d\xi \right] \\ &\leq \lim_{n \rightarrow \infty} \mathbb{E} \left[\int_0^1 -\alpha_y \|\tilde{y}_c^e\|^2 d\xi \right] \\ &= -\alpha_y \|\tilde{y}_c^e\|^2. \end{aligned} \quad (80)$$

In addition, since $K(y^*, \tilde{h}^l) = \mathbf{0}$ due to the property of the stationary point $y^* \in \mathcal{Y}$, we have

$$\begin{aligned} \|k(y_c, \tilde{h}^l)\| &= \lim_{n \rightarrow \infty} \mathbb{E} \|K(y_c, \tilde{h}^l)\| \\ &= \lim_{n \rightarrow \infty} \mathbb{E} \|K(y_c, \tilde{h}^l) - K(y_c^*, \tilde{h}^l)\| \\ &\leq \lim_{n \rightarrow \infty} \mathbb{E} l_y \|y_c - y_c^*\| \\ &= l_y \|\tilde{y}_c^e\| \end{aligned} \quad (81)$$

where the inequality is from the Lipschitz property in Assumption 4.

Finally, using the above result, we can find an upper bound for the Lyapunov drift in (77) as

$$\tilde{\mathcal{L}}V \leq -N_s^{-1}\alpha(\|\tilde{x}_c\|^2 + \|\tilde{y}_c\|^2) + \frac{l_x}{N_s} \|\tilde{y}_c\| \|\tilde{x}_c^e\| - \alpha_x \|\tilde{x}_c^e\|^2 + \frac{1}{2} v_H \|\tilde{x}_c^e\| \|\tilde{h}^s\| \frac{a_H \tau}{N_s \gamma} \quad (82)$$

$$\begin{aligned} &+ \frac{v_y l_y}{N_s} \|\tilde{x}_c^e\| \|\tilde{y}_c^e\| - \frac{\alpha_y}{N_s} \|\tilde{y}_c^e\|^2 + \frac{\epsilon \tau}{N_s \gamma} \varpi \|\tilde{y}_c^e\| - \frac{1}{2} \frac{a_H \tau}{N_s \gamma} \|\tilde{h}^s\|^2 + C_0(y^*(\tilde{h}^l)) \\ &\triangleq \phi(\chi) \end{aligned} \quad (83)$$

where $C_0(y^*(\tilde{h}^l)) = \frac{\tau}{2N_s} \text{tr}(\Sigma(y^*)) + \frac{1}{2} \frac{a_H \tau}{N_s \gamma} v_H^2 N + \frac{1}{2} \frac{a_H \tau}{N_s \gamma} N$ is from the trace terms in (77). The parameter N is the dimension of the parameters h^s and h^l . $\phi : \chi \mapsto \mathbb{R}$ is the *upper bound drift function* and $\chi = (\|\tilde{x}_c\|, \|\tilde{y}_c\|, \|\tilde{x}_c^e\|, \|\tilde{y}_c^e\|, \|\tilde{h}^s\|)$ is the vector measuring the deviations of each virtual state from the origin.

Note that $\phi(\chi)$ is a quadratic function and we can write it as

$$\phi(\chi) = -\chi^T A \chi + b^T \chi + C_0$$

where

$$A = \begin{bmatrix} \frac{\alpha}{N_s} & & \dots & 0 \\ & \frac{\alpha}{N_s} & -\frac{l_x}{2N_s} & \vdots \\ & -\frac{l_x}{2N_s} & \alpha_x & -\frac{v_y l_y}{2N_s} & -\frac{v_H a_H \tau}{4N_s \gamma} \\ \vdots & & -\frac{v_y l_y}{2N_s} & \frac{a_y}{N_s} & 0 \\ 0 & \dots & -\frac{v_H a_H \tau}{4N_s \gamma} & 0 & \frac{1}{2} \frac{a_H \tau}{N_s \gamma} \end{bmatrix} \quad (84)$$

$$b = [0, 0, 0, \frac{\epsilon \omega \tau}{N_s \gamma}, 0]^T \text{ and } C_0 = \frac{1}{2N_s} \left[\tau \text{tr}(\Sigma(y^*)) + \frac{a_H \tau}{\gamma} N(1 + v_H^2) \right].$$

According to Lemma 2, a sufficient condition for the VSDS $u(t)$ to be mean square stable is that the function $\tilde{\mathcal{L}}V(u) \leq \phi(\chi)$ can be further upper bounded by $\phi(\chi) \leq -f(\chi) + C_f$, where $f(\chi) = c\chi^T \chi$, for some constant $c > 0$ and $C_f < \infty$. This is equivalent to verifying if the function $\phi(\chi) + f(\chi) = -\chi^T (A - c\mathbf{I})\chi + b^T \chi + C_0$ is bounded above. Therefore, we only need to check the positive definite property of the coefficient matrix A . To do this, we can calculate each of the leading principle minors of A , and make them positive. These calculations lead to the sufficient condition (39) in Theorem 4.

Remark 6 (The case $y^(\tilde{h}^l)$ on the boundary):* When the optimal solution y^* is on the boundary of the constraint domain, we may have non-zero dZ_y and a non-identity matrix $\Sigma_0(y_c)$. We may have the following two modifications in the above flow. (i) The term $\tilde{y}_c^T [K_x(\cdot)(\tilde{x}_c + \tilde{x}_c^e) + K_y(\cdot)\tilde{y}_c]$ in (77) and (78) now becomes $\tilde{y}_c^T [K_x(\cdot)(\tilde{x}_c + \tilde{x}_c^e) + K_y(\cdot)\tilde{y}_c + dZ_y]$. Since the term about the x variable $K_x(\cdot)(\tilde{x}_c + \tilde{x}_c^e)$ does not contribute to the reflection dZ_y , we only need to evaluate $\tilde{y}_c^T [K_y(\cdot)\tilde{y}_c + dZ_y]$. Note that, when there is a non-zero reflection on the i -th component of y_c , we must have $\tilde{y}_c^{(i)} = 0$ for most of the time. Thus we still have $\tilde{y}_c^T [K_y(\cdot)\tilde{y}_c + dZ_y] \tilde{y}_c \leq -\alpha_y \|\tilde{y}_c\|^2$, as we did in (78). (ii) Consider the covariance matrix in (72). Its trace must be smaller than the case when $y^*(\tilde{h}^l)$ is in the interior, since there are some zero diagonal elements in the matrix $\Sigma_0(y^*)$ [see Appendix D]. Thus the constant C_0 defined above is still an upper bound. Combining the cases (i) and (ii), the optimal solution y^* being on the boundary does not change the upper bound of the drift as in (82). ■

APPENDIX F

PROOF OF THEOREM 5

Choose a function $f(\chi) = c\chi^T A \chi$ for some constant $0 < c < 1$. We have $f(\chi) \geq c\lambda_{\min}(A)\|\chi\|^2$, where $\lambda_{\min}(A)$ denotes the smallest eigenvalue of matrix A .

Under the sufficient condition in Theorem 4 and from (83), we have the stochastic Lyapunov drift be upper bounded by

$$\tilde{\mathcal{L}}V(u) \leq \phi(\chi(u)) = -\chi^T A \chi + b^T \chi + C_0 \leq -f(\chi) + C_1$$

where $C_1 = C_0 + \frac{C_b}{1-c}$, and $C_b = \frac{1}{4}b^T A^{-1}b = \left(\frac{\epsilon \varpi \tau}{N_s \gamma}\right)^2 \frac{4N_s^{-1}\alpha\alpha_x - N_s^{-2}l_x^2 - N_s^{-2}l_y^2 v_y^2}{N_s^{-1}\alpha(8\alpha_x - a_H v_H^2) - 2N_s^{-2}l_x^2 - 2N_s^{-2}l_y^2 v_y^2} < \infty$.

Then, using Lemma 2, we have

$$\begin{aligned} \limsup_{t \rightarrow \infty} \frac{1}{t} \int_0^t \mathbb{E} \|u(\tau)\|^2 d\tau &\leq \frac{1}{c\lambda_{\min}(A)} \times \limsup_{t \rightarrow \infty} \frac{1}{t} \int_0^t \left(C_0 + \frac{C_b}{1-c}\right) d\tau \\ &= \frac{1}{c\lambda_{\min}(A)} \left(\bar{C}_0 + \frac{C_b}{1-c}\right) \end{aligned}$$

where $\bar{C}_0 = \frac{1}{2N_s} \left[\tau \bar{\Sigma} + \frac{a_H \tau}{\gamma} N(1 + v_H^2) \right]$, and $\bar{\Sigma} = \limsup_{t \rightarrow \infty} \frac{1}{t} \int_0^t \text{tr}(\Sigma(y^*(h^l(\tau)))) d\tau$ is the time-averaged covariance matrix of the estimator $K(\cdot)$. Choosing c to minimize the above upper bound, we obtain $c^* = \frac{\bar{C}_0 + C_b}{2\bar{C}_0}$. With the observation that $C_b(\epsilon^2)$ can typically be very small, due to the small timescale separation parameter $\epsilon \ll 1$. Then we choose $c = \frac{1}{2}$ for a reasonable tight upper bound of the tracking error.

We now derive the term $\lambda_{\min}(A)$, by using the eigenvalue lower bound $\lambda_{\min}(A) \geq \frac{|\det(A)|}{2^{n/2-1}\|A\|_F}$ from [44], where $\|\cdot\|_F$ denotes the Frobenius norm and n is the dimension of A . We apply a trick to obtain a good eigenvalue bound by letting $A_0 = N_s A$. Then $\lambda_{\min}(A) = N_s^{-1} \lambda_{\min}(A_0) \geq N_s^{-1} \frac{|\det(A_0)|}{2^{3/2}\|A_0\|_F} = N_s^{-1} \frac{\rho}{\eta}$, where $\rho = |\det(A_0)| = \frac{\alpha a_H \tau}{16\gamma} \left(8N_s \alpha \alpha_x \alpha_y - \alpha \alpha_y \frac{a_H \tau}{\gamma} v_H^2 - 2\alpha_y l_x^2 - 2\alpha l_y^2 v_y^2 \right) = \mathcal{O}(N_s \alpha^2 \alpha_x \alpha_y)$ and $\eta = 2^{3/2} \|A_0\|_F = \mathcal{O}(\sqrt{N_s^2 \alpha_x^2 + \alpha^2})$. Therefore, we have

$$\begin{aligned} \limsup_{t \rightarrow \infty} \frac{1}{t} \int_0^t \mathbb{E} \|u(\tau)\|^2 d\tau &\leq \frac{1}{\frac{1}{2} N_s^{-1} \frac{\rho}{\eta}} \left(\frac{1}{2N_s} \left[\tau \bar{\Sigma} + \frac{a_H \tau}{\gamma} N(1 + v_H^2) \right] + C_b(\epsilon^2) \right) \\ &= \frac{\eta}{\rho} (\tau \bar{\Sigma} + C) \end{aligned}$$

where $C = \frac{a_H \tau}{\gamma} N(1 + v_H^2) + 4C_b(\epsilon) = \frac{a_H \tau}{\gamma} N(1 + v_H^2) + \mathcal{O}(\epsilon^2 \varpi^2 \tau^2 \gamma^{-2})$.

Since $\|\tilde{x}_c\|^2 + \|\tilde{x}_c^e\|^2 + \|\tilde{y}_c\|^2 + \|\tilde{y}_c^e\|^2 \leq \|u\|^2$, we have $e_x + e_y \leq \frac{\eta}{\rho} (\frac{1}{2} \bar{\Sigma} + C)$.

APPENDIX G

PROOF OF THEOREM 6

Using the compensated MCTS (43)-(44) and the CSI dynamics (1)-(2), the dynamic system (45)-(46) can be written as

$$\begin{aligned} d\tilde{x}_c^e &= G(x_c, y_c; \cdot) dt + \left(\varphi_x^h(\cdot) - \hat{\varphi}_x^h(\cdot) \right) \left(-\frac{a_H \tau}{2N_s \gamma} \tilde{h}^s \right) dt + (\varphi_x^y(\cdot) - \hat{\varphi}_x^y(\cdot)) N_s^{-1} k(y_c, \tilde{h}^l) dt \\ &\quad + (\varphi_x^y(\cdot) - \hat{\varphi}_x^y(\cdot)) \hat{\varphi}_y^h(y_c, \tilde{h}^s, \tilde{h}^l) d\tilde{h}^l + \left(\varphi_x^h(\cdot) - \hat{\varphi}_x^h(\cdot) \right) \sqrt{\frac{a_H \tau}{N_s \gamma}} dW_t \end{aligned} \quad (85)$$

$$d\tilde{y}_c^e = N_s^{-1} k(y_c, \tilde{h}^l) dt + \left(\varphi_y^h(\tilde{h}^l) - \hat{\varphi}_y^h(y_c, \tilde{h}^s, \tilde{h}^l) \right) d\tilde{h}^l \quad (86)$$

Consider two Lyapunov functions $V_1(\tilde{x}_c^e) = \frac{1}{2}(\tilde{x}_c^e)^T \tilde{x}_c^e$ and $V_2(\tilde{y}_c^e) = \frac{1}{2}(\tilde{y}_c^e)^T \tilde{y}_c^e$ defined along the trajectory of the virtual system (85)-(86).

Define $\xi_y^h(y_c, \tilde{h}^s, \tilde{h}^l) \triangleq \|\varphi_y^h(\tilde{h}^l) - \hat{\varphi}_y^h(y_c, \tilde{h}^s, \tilde{h}^l)\| - \mathbb{E}\|\varphi_y^h(\tilde{h}^l) - \hat{\varphi}_y^h(y_c, \tilde{h}^s, \tilde{h}^l)\|$. Note that $\xi_y^h(\cdot)$ acts like a “noise” term that depends on \tilde{h}^s and satisfies $\mathbb{E}\xi_y^h(\cdot) = 0$. Using Lemma 3, the Lyapunov drift of $V_2(\cdot)$ is given by

$$\begin{aligned} \tilde{\mathcal{L}}V_2(\cdot) &= (\tilde{y}_c^e)^T N_s^{-1} k(y_c, \tilde{h}^l) + (\tilde{y}_c^e)^T \left(\varphi_y^h(\tilde{h}^l) - \hat{\varphi}_y^h(y_c, \tilde{h}^s, \tilde{h}^l) \right) \frac{d\tilde{h}^l}{dt} \\ &\leq -\frac{\alpha_y}{N_s} \|\tilde{y}_c^e\|^2 + \|\tilde{y}_c^e\| \left(\hat{L}_y^h \|\tilde{y}_c^e\| + \beta_y^h + \xi_y^h(\cdot) \right) \frac{\tau}{N_s \gamma} \epsilon(D_{\min}, v_{\max}) \\ &= -\left(\tilde{\alpha}_y - \tilde{\epsilon} \hat{L}_y^h \right) \|\tilde{y}_c^e\|^2 + \tilde{\epsilon} \beta_y^h \|\tilde{y}_c^e\| + \tilde{\epsilon} \xi_y^h(\cdot) \|\tilde{y}_c^e\| \\ &\leq -c \|\tilde{y}_c^e\| + \tilde{\epsilon} \xi_y^h(\cdot) \|\tilde{y}_c^e\| + \frac{(\tilde{\epsilon} \beta_y^h + c)^2}{4(\tilde{\alpha}_y - \tilde{\epsilon} \hat{L}_y^h)} \end{aligned} \quad (87)$$

where $\epsilon(D_{\min}, v_{\max}) = 2D_{\min}^{-\iota-1} v_{\max}$, $\tilde{\alpha}_y$ and $\tilde{\epsilon} = \frac{\tau}{N_s \gamma} \epsilon$ according to the mobility model and CSI model in Section II-A, $c > 0$ is an arbitrary constant, and the first inequality is from (80).

Notice that $\mathbb{E} \left[\tilde{\epsilon} \xi_y^h(\cdot) \|\tilde{y}_c^e\| + \frac{(\tilde{\epsilon} \beta_y^h + c)^2}{4(\tilde{\alpha}_y - \tilde{\epsilon} \hat{L}_y^h)} \right] = \frac{(\tilde{\epsilon} \beta_y^h + c)^2}{4(\tilde{\alpha}_y - \tilde{\epsilon} \hat{L}_y^h)}$. Using Lemma 2, we obtain the upper bound of the tracking error $\|\tilde{y}_c^e\|$ as

$$\mathbb{E}\|\tilde{y}_c^e\| \leq \frac{(\tilde{\epsilon} \beta_y^h + c)^2}{4c(\tilde{\alpha}_y - \tilde{\epsilon} \hat{L}_y^h)}. \quad (88)$$

A tight upper bound can be given by minimizing (88) over $c > 0$. Then we obtain

$$\mathbb{E}\|\tilde{y}_c^e\| \leq \frac{\tilde{\epsilon} \beta_y^h}{\tilde{\alpha}_y - \tilde{\epsilon} \hat{L}_y^h} = \frac{\epsilon \tau \beta_y^h}{\alpha_y \gamma - \epsilon \tau \hat{L}_y^h}. \quad (89)$$

To study $V_1(\cdot)$, define $\xi_h = \|\tilde{h}^s\| - \mathbb{E}\|\tilde{h}^s\|$, where $\mathbb{E}\|\tilde{h}^s\| = \sqrt{\frac{2}{\pi}}$ according to the CSI model in (1). Define $\xi_y = \|\tilde{y}_c^e\| - \mathbb{E}\|\tilde{y}_c^e\|$. Note that ξ_h and ξ_y only depend on \tilde{h}^s and \tilde{y}_c^e , respectively, and $\mathbb{E}\xi_h = \mathbb{E}\|\tilde{y}_c^e\| = 0$.

We then derive the Lyapunov drift of $V_1(\cdot)$ as follows,

$$\begin{aligned} \tilde{\mathcal{L}}V_1(\cdot) &\approx (\tilde{x}_c^e)^T G(\cdot) + (\tilde{x}_c^e)^T \left(\varphi_x^h(\cdot) - \hat{\varphi}_x^h(\cdot) \right) \left(-\frac{a_H \tau}{2N_s \gamma} \tilde{h}^s \right) + (\tilde{x}_c^e)^T (\varphi_x^y(\cdot) - \hat{\varphi}_x^y(\cdot)) N_s^{-1} k(y_c, \tilde{h}^l) \\ &\quad + (\varphi_x^y(\cdot) - \hat{\varphi}_x^y(\cdot)) \hat{\varphi}_y^h(y_c, \cdot) d\tilde{h}^l + \text{tr} \left[\left(\varphi_x^h(\cdot) - \hat{\varphi}_x^h(\cdot) \right) \frac{a_H \tau}{N_s \gamma} \left(\varphi_x^h(\cdot) - \hat{\varphi}_x^h(\cdot) \right)^T \right] \\ &\leq -\alpha_x \|\tilde{x}_c^e\|^2 + \hat{L}_x^h \|\tilde{x}_c^e\|^2 \frac{a_H \tau}{2N_s \gamma} \left(\sqrt{\frac{2}{\pi}} + \xi_h \right) + \hat{L}_x^y \|\tilde{x}_c^e\|^2 N_s^{-1} l_y (\mathbb{E}\|\tilde{y}_c^e\| + \xi_y) \\ &\quad + \frac{a_H \tau}{N_s \gamma} \left(\hat{L}_x^h \right)^2 \|\tilde{x}_c^e\|^2 \end{aligned} \quad (90)$$

$$\begin{aligned} &\leq -\left(\alpha_x - \frac{a_H \tau}{\sqrt{2\pi} N_s \gamma} \hat{L}_x^h - \hat{L}_x^y N_s^{-1} l_y \frac{\epsilon \tau \beta_y^h}{\alpha_y \gamma - \epsilon \tau \hat{L}_y^h} - \frac{a_H \tau}{N_s \gamma} \left(\hat{L}_x^h \right)^2 \right) \|\tilde{x}_c^e\|^2 \\ &\quad + \hat{L}_x^h \|\tilde{x}_c^e\|^2 \frac{a_H \tau}{2N_s \gamma} \xi_h + \hat{L}_x^y \|\tilde{x}_c^e\|^2 l_y \xi_y \end{aligned} \quad (91)$$

(c.f. equations (79) and (81) for the first inequality), where in (90), the $d\tilde{h}^l$ term is dropped, since $d\tilde{h}^l$ is much smaller than the \tilde{h}^s .

Note that the last two terms in (91) have mean 0 due to the “noise” terms ξ_h and ξ_y . Therefore, according to Lemma 2, if $\alpha_x - \frac{a_H \tau}{\sqrt{2\pi} N_s \gamma} \hat{L}_x^h - \hat{L}_x^y N_s^{-1} l_y \frac{\epsilon \tau \beta_y^h}{\alpha_y \gamma - \epsilon \tau \hat{L}_y^h} - \frac{a_H \tau}{N_s \gamma} \left(\hat{L}_x^h \right)^2 > 0$, then \tilde{x}_c^e converges to 0 in probability.

APPENDIX H

PROOF OF THEOREM 7

We have the iteration mappings $G(\cdot) = \left(\frac{\partial}{\partial \mathbf{p}} L(\cdot), -\frac{\partial}{\partial \lambda} L(\cdot) \right)$ and $K(\cdot) = \frac{\partial}{\partial \mathbf{r}} L(\cdot)$. In addition, we have $G_x(\cdot) = M_L(\cdot) = \begin{pmatrix} \frac{\partial^2}{\partial \mathbf{p} \partial \mathbf{p}} L(\cdot) & \frac{\partial^2}{\partial \mathbf{p} \partial \lambda} L(\cdot) \\ -\frac{\partial^2}{\partial \lambda \partial \mathbf{p}} L(\cdot) & \mathbf{0}_{J \times J} \end{pmatrix}$ and $K_y(\cdot) = \frac{\partial^2}{\partial \mathbf{r} \partial \mathbf{r}} L(\cdot)$. From the convex assumption, $K_y(\cdot)$ is negative definite. It follows that $\alpha_y(\cdot) = -\lambda_{\max} \left(\frac{\partial^2}{\partial \mathbf{r} \partial \mathbf{r}} L(\cdot) \right)$.

Note that, the Hessian of the Lagrange function $\frac{\partial^2}{\partial \mathbf{p} \partial \mathbf{p}} L(\cdot)$ is negative definite. Then according to [32, Proposition 4.4.2], the matrix $M_L(\cdot)$ is also negative definite.

To derive the convergence for the x part, we consider a Lyapunov function $V(x_e) = \frac{1}{2} x_e^T x_e$, where $x_e = x - x^*$. From a modification of Ito’s formula in Lemma 3, it follows that $\tilde{\mathcal{L}}V = \frac{1}{2} (x_e^T dx_e + dx_e^T x_e) = \frac{1}{2} (x_e^T G(\cdot) + G(\cdot)^T x_e) \leq \frac{1}{2} (x_e^T G_x x_e + x_e^T G_x x_e) = \frac{1}{2} x_e^T (G_x + G_x^T) x_e \leq -\lambda_{\max}(\frac{1}{2} (G_x + G_x^T))$.

REFERENCES

- [1] L. Chen, S. Low, M. Chiang, and J. Doyle, “Cross-layer congestion control, routing and scheduling design in ad hoc wireless networks,” 2006.
- [2] L. Georgiadis, M. J. Neely, and L. Tassiulas, “Resource allocation and cross-layer control in wireless networks,” *Now Publishers*, vol. 1, pp. 1–144, April 2006.
- [3] D. P. Palomar, J. M. Cioffi, and M. A. Lagunas, “Joint tx-rx beamforming design for multicarrier mimo channels: A unified framework for convex optimization,” *IEEE Transactions on Signal Processing*, vol. 51, no. 9, pp. 2381–2401, 2003.
- [4] L. Zhang, Y.-C. Liang, and Y. Xin, “Joint beamforming and power allocation for multiple access channels in cognitive radio networks,” *IEEE Journal on Selected Areas in Communications*, vol. 26, no. 1, pp. 38–51, 2008.
- [5] S. Ghosh, B. D. Rao, and J. R. Zeidler, “Scheduling and power control in statistical beamforming networks using b bits of feedback,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) 2012*. IEEE, 2012, pp. 2797–2800.
- [6] I. Wajid, Y. C. Eldar, and A. Gershman, “Robust downlink beamforming using covariance channel state information,” in *IEEE International Conference on Acoustics, Speech and Signal Processing, 2009. ICASSP 2009*. IEEE, 2009, pp. 2285–2288.
- [7] N. Bornhorst and M. Pesavento, “Beamforming for multi-group multicasting with statistical channel state information using second-order cone programming,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2012*. IEEE, 2012, pp. 3237–3240.
- [8] T. S. Feng, T. R. Field, and S. Haykin, “Stochastic Differential Equation Theory Applied to Wireless Channels,” *IEEE Transactions on Communications*, vol. 55, no. 8, pp. 1478–1483, Aug. 2007.

- [9] Y. Mostofi, A. Gonzalez-Ruiz, A. Gaffarkhah, and D. Li, "Characterization and modeling of wireless channels for networked robotic and control systems-a comprehensive overview," in *IEEE/RSJ International Conference on Intelligent Robots and Systems, 2009. IROS 2009*. IEEE, 2009, pp. 4849–4854.
- [10] M. Goyal, A. Kumar, and V. Sharma, "Power constrained and delay optimal policies for scheduling transmission over a fading channel," in *IEEE INFOCOM 2003*, vol. 1, march-3 april 2003, pp. 311 – 320 vol.1.
- [11] M. J. Neely, "Order optimal delay for opportunistic scheduling in multi-user wireless uplinks and downlinks," *IEEE/ACM Transactions on Networking*, vol. 16, pp. 1188–1199, October 2008. [Online]. Available: <http://dx.doi.org/10.1109/TNET.2007.909682>
- [12] M. Chiang, "Balancing transport and physical Layers in wireless multihop networks: jointly optimal congestion control and power control," *IEEE Journal on Selected Areas in Communications*, vol. 23, no. 1, pp. 104–116, 2005.
- [13] J. Gomez and A. Campbell, "Supporting adaptive-qos over multiple time scales in wireless networks," *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, 1999.
- [14] H. Lin, W. Wu, Y. Ren, and X. Shan, "A time-scale decomposition approach to optimize wireless packet resource allocation and scheduling," in *IEEE Wireless Communications and Networking Conference, 2002. WCNC2002*, vol. 2. IEEE, 2002, pp. 699–705.
- [15] J. Papandriopoulos, S. Dey, and J. Evans, "Optimal and distributed protocols for cross-layer design of physical and transport layers in manets," *IEEE/ACM Transactions on Networking (TON)*, vol. 16, no. 6, pp. 1392–1405, 2008.
- [16] P. Soldati and M. Johansson, "Reducing signaling and respecting time-scales in cross-layer protocols design for wireless networks," in *Global Telecommunications Conference, 2009. IEEE GLOBECOM 2009*. IEEE, 2009, pp. 1–8.
- [17] X. Zheng, F. Chen, Y. Xia, and Y. Fang, "A class of cross-layer optimization algorithms for performance and complexity trade-offs in wireless networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 20, no. 10, pp. 1393–1407, 2009.
- [18] M. Benaïm, "Dynamics of stochastic approximation algorithms," *Seminaire de probabilites XXXIII*, pp. 1–68, 1999.
- [19] H. Kushner and G. Yin, *Stochastic approximation and recursive algorithms and applications*. Springer, 2003, vol. 35.
- [20] J. Birge and F. Louveau, *Introduction to stochastic programming*. Springer Verlag, 1997.
- [21] T. Santoso, S. Ahmed, M. Goetschalckx, and A. Shapiro, "A stochastic programming approach for supply chain network design under uncertainty," *European Journal of Operational Research*, vol. 167, no. 1, pp. 96–115, 2005.
- [22] A. Shapiro, D. Dentcheva, and A. Ruszczyński, *Lectures on stochastic programming: modeling and theory*. Society for Industrial Mathematics, 2009, vol. 9.
- [23] J. Chen and V. Lau, "Convergence analysis of saddle point problems in time varying wireless systems - control theoretical approach," *IEEE Transactions on Signal Processing*, vol. 60, no. 1, pp. 443–452, January 2012.
- [24] —, "Delay analysis of max-weight queue algorithm for time-varying wireless ad hoc networks - control theoretical approach," *IEEE Transactions on Signal Processing*, vol. 61, no. 1, pp. 99–108, January 2013.
- [25] G. Yin and Q. Zhang, *Discrete-time Markov chains: two-time-scale methods and applications*. Springer, 2004, vol. 55.
- [26] G. Yin, Q. Zhang, J. Moore, and Y. Liu, "Continuous-time tracking algorithms involving two-time-scale markov chains," *IEEE Transactions on Signal Processing*, vol. 53, no. 12, pp. 4442–4452, 2005.
- [27] A. Costa and F. Vázquez-Abad, "Adaptive stepsize selection for tracking in a regime-switching environment," *Automatica*, vol. 43, no. 11, pp. 1896–1908, 2007.
- [28] N. Bansal and Z. Liu, "Capacity, delay and mobility in wireless ad-hoc networks," in *INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications. IEEE Societies*, vol. 2. IEEE, 2003, pp. 1553–1563.
- [29] K. Lee, S. Hong, S. J. Kim, I. Rhee, and S. Chong, "Slaw: A new mobility model for human walks," in *INFOCOM 2009, IEEE*. IEEE, 2009, pp. 855–863.

- [30] I. Rhee, M. Shin, S. Hong, K. Lee, S. J. Kim, and S. Chong, "On the levy-walk nature of human mobility," *IEEE/ACM Transactions on Networking (TON)*, vol. 19, no. 3, pp. 630–643, 2011.
- [31] D. Tse and P. Viswanath, *Fundamentals of Wireless Communication*. Cambridge University Press, 2005.
- [32] D. Bertsekas, "Nonlinear programming," 1999.
- [33] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.
- [34] M. Chiang, S. H. Low, A. R. Calderbank, and J. C. Doyle, "Layering as Optimization Decomposition: A Mathematical Theory of Network Architectures," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 255–312, January 2007. [Online]. Available: <http://dx.doi.org/10.1109/JPROC.2006.887322>
- [35] G. Tychogiorgos, A. Gkelias, and K. Leung, "Utility-proportional fairness in wireless networks," in *2012 IEEE 23rd International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC)*. IEEE, 2012, pp. 839–844.
- [36] A. Benveniste, P. Priouret, and M. Métivier, *Adaptive algorithms and stochastic approximations*. Springer-Verlag New York, Inc., 1990.
- [37] J. Zhang, D. Zheng, and M. Chiang, "Impact of stochastic noisy feedback on distributed network utility maximization," *IN INFOCOM 2007*, vol. 2008, pp. 222–230, 2007.
- [38] A. V. Skorohod, "Stochastic equations for diffusion processes in a bounded region," *Theory of Probability and Its Applications*, vol. 6, pp. 264–274, 1961.
- [39] H. K. Khalil, *Nonlinear Systems*. Prentice-Hall, 1996.
- [40] S. P. Meyn and R. L. Tweedie, "Stability of markovian processes III: Foster-Lyapunov criteria for continuous-time processes," *Advances in Applied Probability*, vol. 25, no. 3, pp. 518–548, Sept 1993.
- [41] S. Boyd, L. El Ghaoul, E. Feron, and V. Balakrishnan, *Linear matrix inequalities in system and control theory*. Society for Industrial Mathematics, 1987, vol. 15.
- [42] "Requirements for eutra and eutran," 3GPP TR 25.913, Tech. Rep.
- [43] R. Buche and H. J. Kushner, "Rate of convergence for constrained stochastic approximation algorithms," *SIAM journal on control and optimization*, vol. 40, no. 4, pp. 1011–1041, 2002.
- [44] G. Piazza and T. Politi, "An upper bound for the condition number of a matrix in spectral norm," *Journal of Computational and Applied Mathematics*, vol. 143, no. 1, pp. 141–144, 2002.